Instituto Tecnológico de Costa Rica

Área Académica de Administración de Tecnologías de Información



Curso:

TI-1401 Taller de Programación

Segundo Proyecto Programado

Tema:

Tokenización de Cadenas

Profesor:

Ing. Luis Pablo Soto Chaves

Estudiantes:

José Manuel Altamirano Salazar - 2020426159

Josué Brenes Alfaro - 2020054427

Fecha de Entrega:

15 de julio de 2020

**Índice**

[Introducción 2](#_Toc45373500)

[Estrategia Metodológica 3](#_Toc45373501)

[Asignación de roles y tareas 3](#_Toc45373502)

[Mecanismos de Seguimiento 4](#_Toc45373503)

[Reuniones y validación de avances 5](#_Toc45373504)

[Modalidad de Trabajo 5](#_Toc45373505)

[Manual de usuario 7](#_Toc45373506)

[Introducción: 8](#_Toc45373507)

[Funcionalidad General: 9](#_Toc45373508)

[Requisitos del Programa: 11](#_Toc45373509)

[Guía de Uso: 14](#_Toc45373510)

[Pruebas de Funcionalidad 34](#_Toc45373511)

[Ventana principal 34](#_Toc45373512)

[Lecciones Aprendidas 48](#_Toc45373513)

[Bitácora de Trabajo 51](#_Toc45373514)

[Análisis de Resultados 52](#_Toc45373515)

[Objetivos Alcanzados 52](#_Toc45373516)

[Objetivos No Alcanzados 52](#_Toc45373517)

[Análisis de Resultados 53](#_Toc45373518)

# Introducción

El análisis de un texto puede darse de diferentes maneras. Puede que la necesidad de estudiar su contenido se base solamente en determinar su estructura o composición. Quizá su estudio sea solo el punto de partida para poder realizar otras observaciones que permitan comprender tanto la forma correcta de escribir como la manera en que ordenamos y comunicamos nuestros pensamientos. Cualquiera que sea el requisito que se deba cumplir en esta línea, el primer paso sería conocer la manera en que se encuentra constituido el documento sobre el cual se esté trabajando.

Con base en lo anterior, es que se apoya la funcionalidad que puede encontrarse en el Programa para Tokenización de Texto que debe desarrollar como parte del segundo proyecto programado, el cual se describe en este documento. La intención principal de este programa es ofrecer una herramienta para la clasificación de los componentes de las oraciones y su presentación estructurada, para que represente un insumo valioso para quien lo necesite.

A través de los requerimientos planteados para este proyecto, se pretende también ampliar el conocimiento en relación con la administración de archivos dentro del lenguaje de programación Python. Al poder crear, extraer contenido o modificar los diferentes tipos de archivos, será posible ampliar la gama de entradas para las soluciones programáticas que deban desarrollarse en el futuro.

Otra área importante de exploración dentro del proyecto será el desarrollo de interfaces gráficas para presentar información al usuario de la solución mediante esta modalidad. Con esta aproximación buscará proporcionar un medio más amigable para interactuar con las diferentes funcionalidades del programa, así como atractivo visualmente.

Para el desarrollo de la solución, se emplearán diferentes tipos de algoritmos secuenciales, iterativos y recursivos, para alcanzar la funcionalidad requerida. Muchos permitirán la manipulación del texto recibido, su separación, etiquetado, traducción al idioma inglés, y posteriormente su almacenamiento en el medio dispuesto para este propósito, que en este caso corresponderá a un archivo de tipo HTML.

# Estrategia Metodológica

## Asignación de roles y tareas

Para la realización del proyecto, se determinó que la forma más conveniente de trabajar cada uno de sus elementos era a partir de la división de sus componentes en una serie de actividades. En ese sentido, se planteó que la división de estas como Actividades Generales o de Revisión, Actividades de Desarrollo de Código y Actividades de Documentación. Dentro de cada uno de estos apartados incluyen cada una de las tareas necesarias para poder completarlo, que además refieren a cada uno de los puntos a ser evaluados como parte del proyecto.

Ahora bien, con el fin de detallar un poco más la manera como serán estructuradas las tareas y roles, conviene mencionar algunas de las divisiones comprendidas por cada segmento. En el caso de las Actividades Generales, serán incluidas aquellas tareas que comprendan labores comunes a todos los miembros del equipo de trabajo. Algunas de estas incluyen el registro de los integrantes del grupo, las revisiones y reuniones periódicas sobre el desarrollo de los otros apartados, las tareas de investigación sobre funciones y librerías, y al culminar el proyecto, la evaluación del trabajo realizado. Para este bloque la asignación de las tareas se da de forma proporcionada en su mayoría, incluyendo a todos los miembros del equipo para su realización.

Por otra parte, entre las tareas comprendidas por el segmento de Actividades de Desarrollo de Código, pueden hallarse todas las tareas de codificación de las funciones que forman parte de la lógica del negocio, la validación y asignación de las restricciones de cada sección y de la estructuración de la interfaz gráfica, requerida para el uso del programa principal. En este caso, la asignación de las tareas se da forma equitativa, asignando de forma combinada la codificación de algunas funciones de la funcionalidad interna y las validaciones y también las requeridas para la interfaz gráfica.

Para terminar, en el caso de las Actividades de Documentación, corresponde incluir todas aquellas tareas que involucren el registro de pruebas y procesos, toma de decisiones, cambios o ajustes que deban realizarse, así como la elaboración del Manual de Usuario para el programa a desarrollar. Otras actividades que se incluyen en este apartado son la publicación de entradas periódicas en el blog de cada integrante del equipo y la elaboración del presente documento. La asignación de las tareas de este segmento se da tanto de forma conjunta como individual, esto de acuerdo con la naturaleza del trabajo que de realizarse para cada una y del tiempo que podría tomar su realización.

## Mecanismos de Seguimiento

El seguimiento del proyecto se llevará a cabo a partir de dos mecanismos principalmente. El primero de estos refiere a la forma en cómo se buscará editar e integrar el código para su desarrollo y revisión. En este caso se hará uso de la herramienta GitHub, con la cual, por medio de la creación de un repositorio será posible almacenar los archivos de código fuente para que todos los integrantes del equipo puedan disponer de estos, realizar los cambios que sean necesarios y completar cada una de sus asignaciones de forma rápida y sencilla.

Junto al mecanismo mencionado, se dispone la aplicación GitHub Desktop, la cual permite sincronizar los archivos disponibles en el repositorio con una carpeta de forma local en el equipo de cada integrante. Con esta función, es posible manejar un respaldo del desarrollo que se tenga hasta el momento. Adicionalmente la plataforma GitHub y su aplicación permiten la creación de “ramas” o subdivisiones del programa o código principal tantas veces como se necesario, y de esta esta manera es posible trabajar de forma separada secciones del código que podrían afectar o intervenir las áreas en las que otro integrante esté trabajando.

Así mismo, la intención de utilizar esta herramienta es llevar un control adicional sobre las actividades que se estén desarrollando en un momento determinado, y que cuando estas sean completadas, sea posible llevar a cabo una integración del código desarrollado de forma separada sin mayor problema.

El segundo mecanismo por utilizar será la ubicación de la documentación externa al código del programa, dentro de una carpeta en la plataforma de Google Drive. Esta permitirá la creación y posteriormente la edición de los documentos de forma directa, sincronizada e incluso colaborativa.

La funcionalidad que ofrece esta herramienta sobre la edición paralela de los archivos de texto y hojas de cálculo proporciona la facilidad de editar los documentos de pruebas del programa o de seguimiento de las actividades de forma sencilla y segura, y que estos cambios puedan ser revisados por los demás integrantes de forma no solo sobre la forma en que se hicieron sino también de acuerdo con el momento en que se dieron.

Finalmente, la creación de la carpeta mencionada anteriormente habilitará un espacio para la colocación de todos los recursos, referencias y material de apoyo tanto para la elaboración de la documentación, como para la codificación de la solución programada.

## Reuniones y validación de avances

En lo que respecta a las reuniones o sesiones de revisión del progreso del proyecto, se acordó que lo ideal sería realizar al menos dos sesiones semanales, los miércoles, jueves o domingo. De esta manera se dispondría del tiempo necesario para trabajar en su desarrollo y validar el estado de cada tarea, y en caso de fuese necesario, realizar las consultas correspondientes en los espacios destinados para este fin durante la semana.

## 

## Modalidad de Trabajo

La modalidad de trabajo que se emplea será con un enfoque combinada, en el cual se tendrán sesiones de trabajo individuales y asincrónicas para las tareas que así lo requieran, así como sesiones remotas de forma sincrónica como es el caso de las reuniones o revisiones del progreso. Para las sesiones remotas se hará uso de plataformas de videoconferencias o videollamadas como lo son Zoom, Google Meet, Discord, entre otras.

# Cronograma de Trabajo

Enlace de acceso al cronograma en Google Drive para facilitar su lectura: <https://bit.ly/CronogramaPP2>

Manual de Usuario

del

Programa para

Tokenización de Texto

# Manual de usuario

TI-1401 Taller de Programación

Segundo Proyecto Programado

I Semestre, 2020

Integrantes:

Jose Altamirano Salazar - 2020426159

Josué Brenes Alfaro -2020054427

## Introducción:

El Programa para Tokenización de Texto, se presenta como una herramienta para iniciar el proceso de análisis de un texto. Esto se realiza a través de la clasificación de los componentes de las oraciones y su presentación estructurada. Partiendo de un proceso conocido como tokenización, que se encargará de categorizar cada una de las palabras según sea un artículo, preposición, pronombre, verbo número o que no pertenezca a ninguna de las anteriores.

Además, cuenta con la funcionalidad para ingresar texto de diferentes maneras. El usuario podrá ingresarlo de forma manual o a través de la lectura de archivos de texto que se encuentren en su computador.

Por otra parte, el programa brinda al usuario la capacidad de traducir algunos de los elementos que integren el texto. Esta operación le ofrece una comprensión mucho más amplia sobre el contenido del documento, al proporcionarle un entendimiento de algunas de las partes del texto incluso dentro del idioma inglés. Esta traducción puede ser consultada desde el programa o desde los archivos generados por el este.

Adicional a las funciones mencionadas, el programa cuenta con la utilidad de generar un archivo HTML que contiene la estructuración del texto generada dentro del programa. Esta función sobre la portabilidad permite al usuario almacenar los tokens extraídos para poder consultarlos posteriormente. La flexibilidad que ofrecen los HTML al poder interpretarse dentro de cualquier sistema operativo, al contar solamente con un navegador web, le permite al usuario acceder a los archivos generados cuando y donde lo necesite.

La ventana principal tiene un enfoque visual atractivo al usuario, con una funcionalidad intuitiva, para que cualquier persona pueda usarlo. La paleta de colores empleada para el diseño utiliza una combinación que no se convierte en un inconveniente a la vista para navegar dentro del programa. La interacción con el programa se realiza por medio de un menú de opciones en la parte superior y una serie botones, los cuales indican su propósito en la leyenda que contienen.

## Funcionalidad General:

La funcionalidad principal del programa de tokenización se centra en realizar la clasificación de un texto, según el origen morfológico de las palabras que integran cada oración recibida. Con esto se hace referencia a la categorización derivado de realizar un análisis morfológico de un texto. En el caso del programa esta labor la realiza por medio de la separación de cada elemento de del texto recibido en partículas únicas denominadas tokens, a través de un proceso de comparación de cada una de estas con sus posibles valores dentro de las categorías delimitadas.

En este caso, las categorías mencionadas corresponden a las siguientes:

* Artículos: el, la, los, las, un, una, unos, unas, lo, del y al.
* Preposiciones: a, ante, bajo, cabe, con, contra, de, desde, durante, en, entre, hacia, hasta, mediante, para, por, según, sin, so, sobre, tras, versus, vía
* Pronombres: yo, me, mí, conmigo, nosotros, nosotras, nos, tú, te, ti, contigo, vosotros, vosotras, vos, él, ella, se, consigo, le, les, mío, mía, míos, mías, nuestro, nuestra, nuestros, nuestras, tuyo, tuya, tuyos, vuestro, vuestra, vuestros, vuestras, suyo, suya, suyos, suyas.
* Formas Verbales / Verbos:
  + Infinitivos: verbos terminados en ar, er, ir
  + Gerundio: verbos terminados en ando, iendo
  + Participio: verbos terminados en ado, ido, to, so, cho
* Números: en su representación numérica de base 10 (0, 1, 2, 3, ... 9)
* Sin Clasificar: cualquier otro token no identificado en las categorías anteriores.

De esta manera, en caso de que alguna de las palabras extraídas del texto ingresado cumpla con las características propias de alguna de las categorías anteriores, esta será incluida en una lista y posteriormente desplegada para ser visualizada por el usuario del programa.

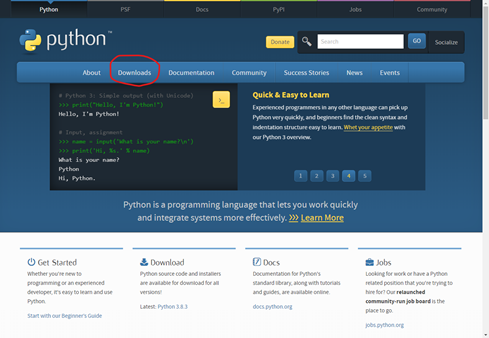
Otra función del programa corresponde la clasificación y traducción de algunos de los tokens del texto recibido. En este caso se realiza la traducción de los que correspondan a las categorías de artículos, pronombres preposiciones y verbos.

Finalmente, el programa también dispone de la posibilidad de almacenar la clasificación generada dentro de un archivo de tipo HTML. Dentro de este archivo se incluye una tabulación de cada una de las categorías previamente mencionadas, para que esta información pueda ser transportada o consultada sin necesidad de volver a procesar un texto en específico.

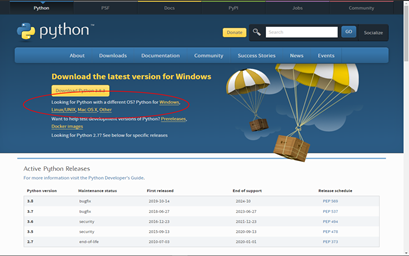
## 

## Requisitos del Programa:

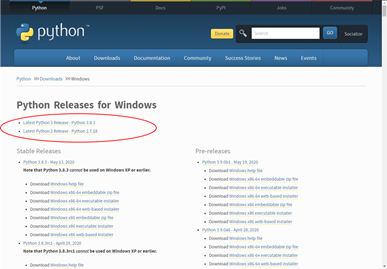
Para poder correr el programa debe tener un intérprete de Python. El programa se encuentra desarrollado bajo la versión 3.8, por lo cual es necesario descargar cualquier versión que iguale o supere a esta. Esta descarga se realiza por medio de la página oficial de Python (<https://www.python.org/>). Al ingresar a esta, es necesario navegar a la opción de descargas en la barra de menú superior.



Al ingresar a esta sección, es necesario que seleccione el sistema operativo de su equipo para realizar la descarga apropiada.

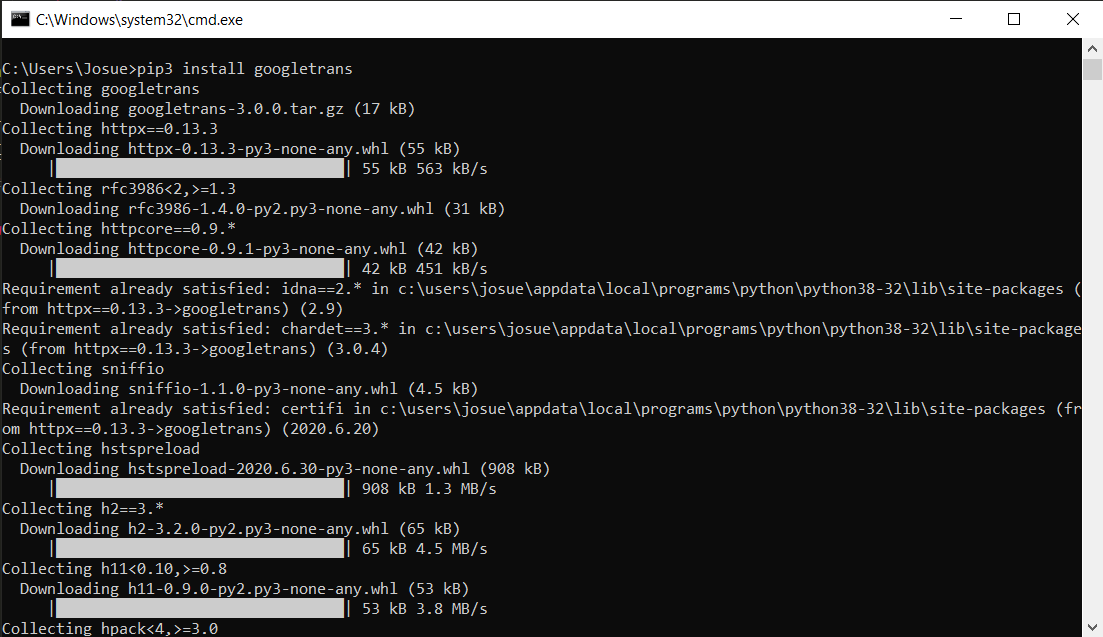


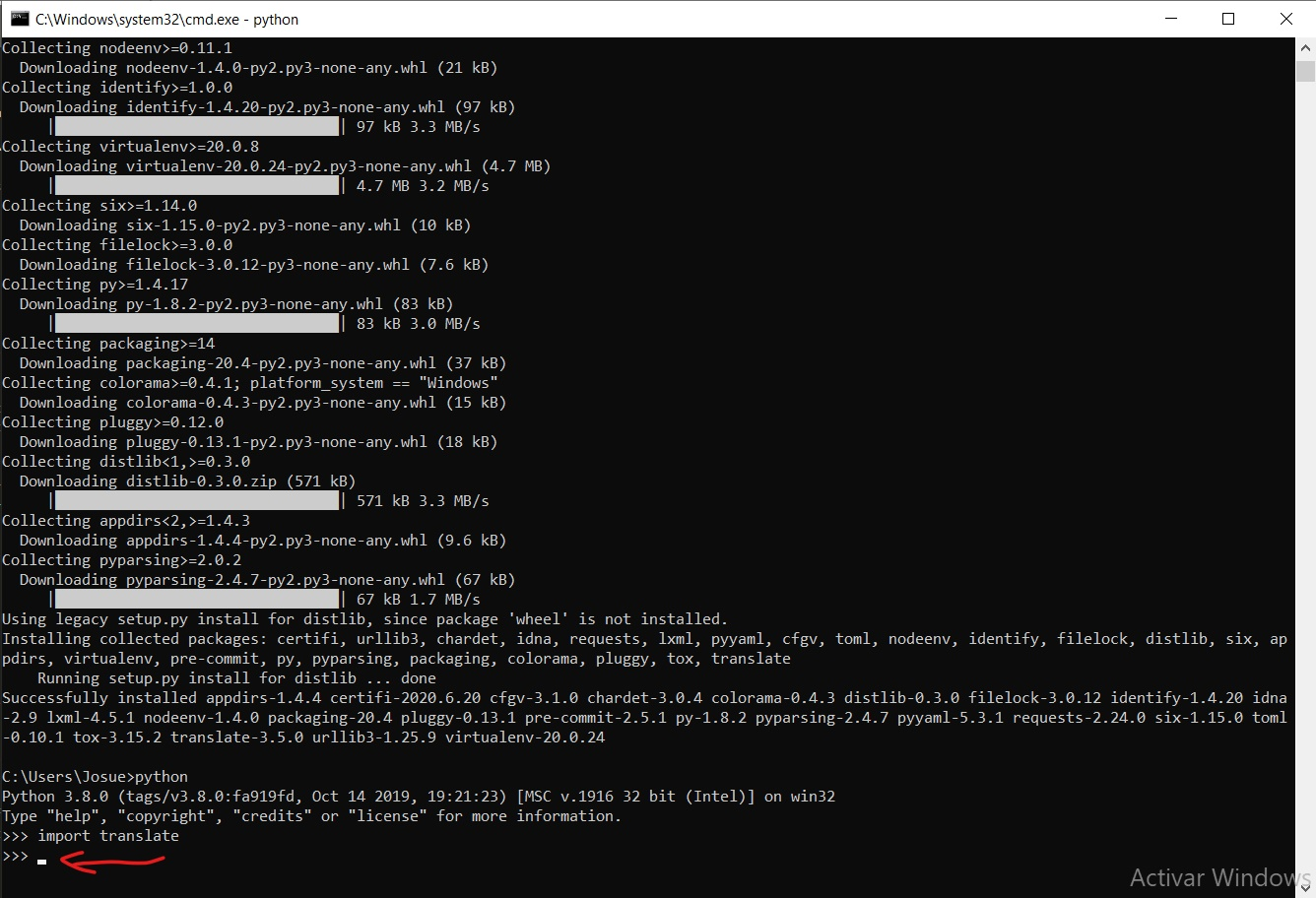
En cualquiera de los casos debe seleccionar la opción que coincida o supera la versión 3.8. Generalmente se incluye en la parte superior de las descargas de cada sistema operativos, tal como se muestra en la siguiente imagen para el caso de Windows.



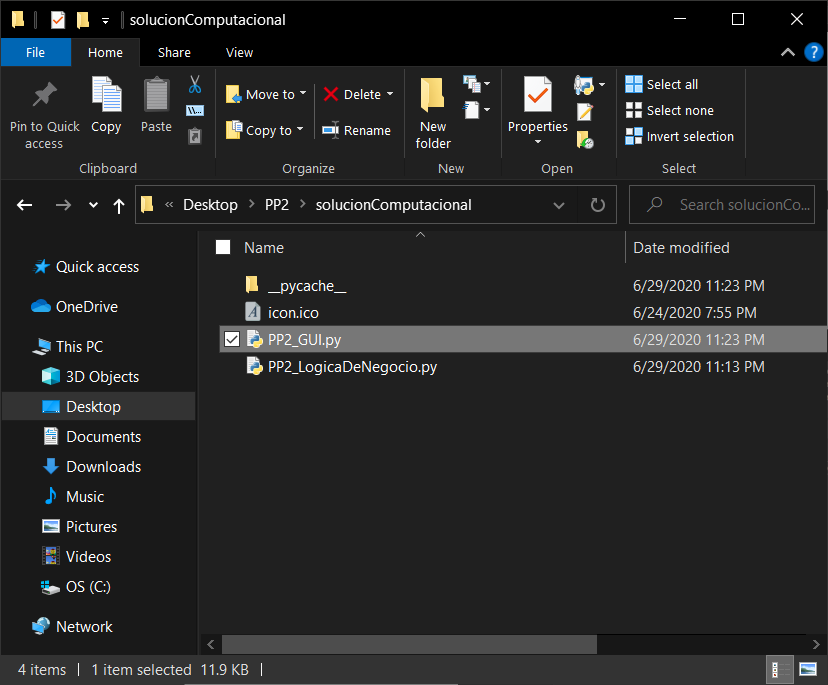
Una vez que el ejecutable de Python se haya descargado proceda a instalarlo. Cuando esta instalación haya finalizado, será necesario instalar la librería para el correcto funcionamiento del programa. La primera es la librería **“Translator”,** para extraer lafuncionalidad del API de traducción de Google, *googletrans*. Para instalar abra una ventana de la consola de Windows. Esto puede hacerlo desde el menú de inicio, buscando el programa CMD o con la combinación de las teclas “Windows” y la tecla R, y luego escribiendo CMD en la ventana que se despliega.

Cuando se ejecute la consola de Windows, deberá ingresar el comando **pip3 install googletrans**. y presionar la tecla Enter. luego de esto se iniciará a instalación de la librería. Al finalizar se desplegará la línea para ingresar un nuevo comando en caso de que no haya sucedido ningún inconveniente.





Ahora diríjase a la ubicación donde haya almacenado el archivo ejecutable del programa de Tokenización. Al hacer doble clic sobre el archivo **“PP2\_GUI.py”**, se iniciará su ejecución. La siguiente imagen muestra este resultado dentro del SO Windows.

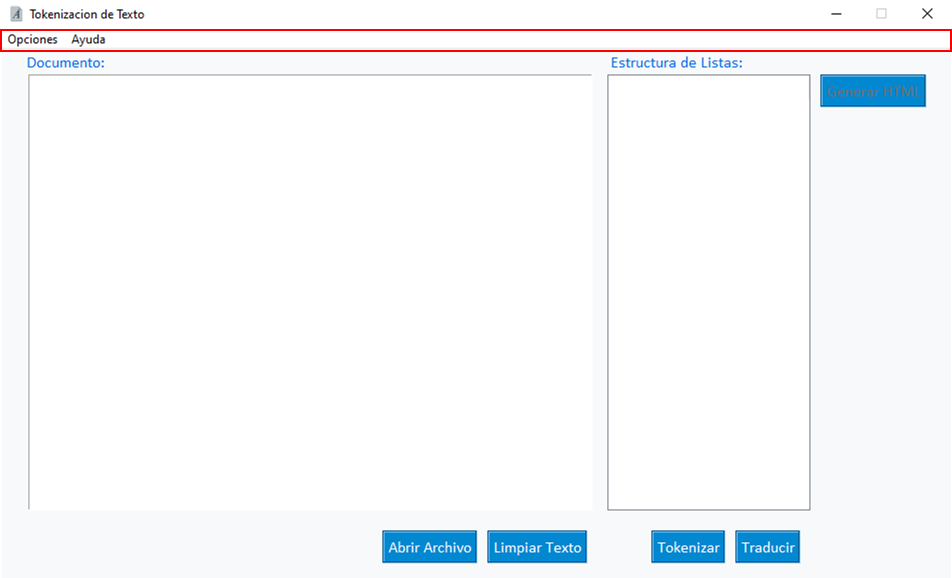


## Guía de Uso:

El programa de Tokenización se compone de una única ventana principal y de algunos mensajes emergentes para indicar al usuario sobre el estado de los procesos o tareas que se estén desempeñando. Para una mejor comprensión sobre las secciones de la ventana y de las opciones de las cuales dispone el programa se ofrece el siguiente desglose explicativo.

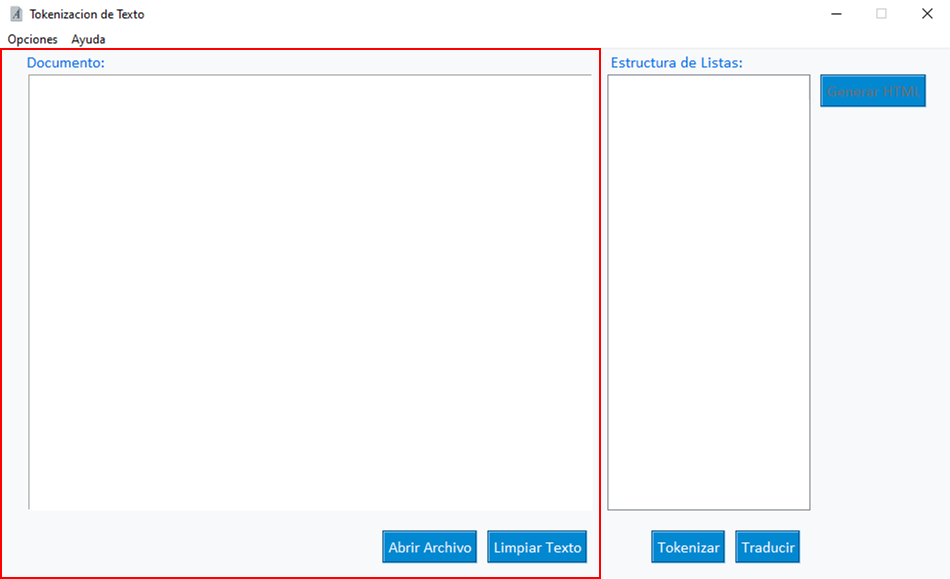
* **Secciones del Programa**

La ventana principal del programa puede dividirse en tres secciones principales. La primera corresponde a la barra de menú. Esta se encuentra en la parte superior de la pantalla, permite acceder a algunas funciones del programa, solicitar una guía sobre el uso de este, o cerrarlo. Su funcionalidad y secciones se explicarán de una forma más amplia en el siguiente apartado.

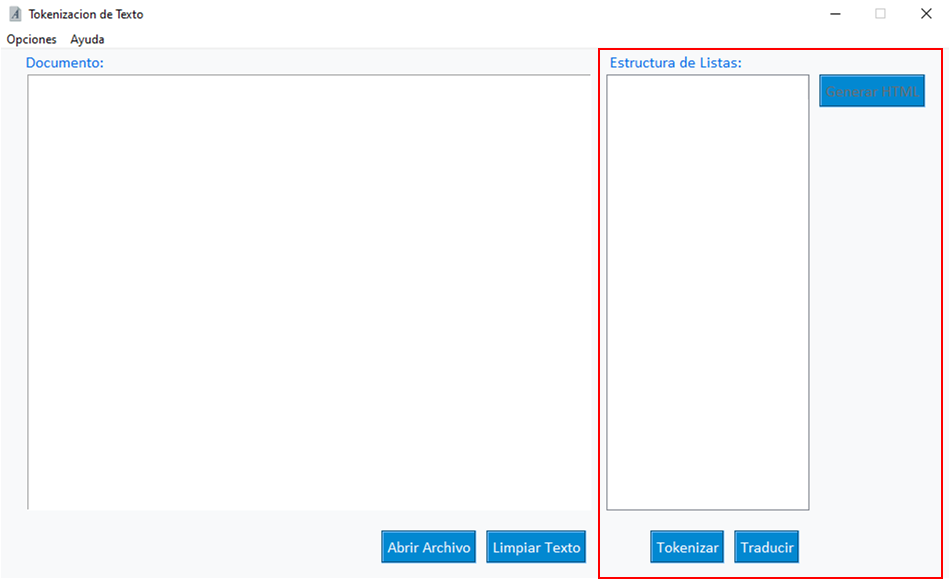


La segunda sección corresponde a la de **“Documento”,** en esta se realiza ingreso y la edición de texto. Se ubica en el lado izquierdo de la ventana principal y contiene principalmente un campo de texto. En este es posible ingresar texto de forma manual desde el teclado del computador o copiando este desde otra ventana. Es también en este espacio donde se mostrará el texto que está contenido en los archivos de texto que sean requeridos y abiertos dentro del programa.

Aun dentro de la sección de **“Documento”**, en la parte inferior se encuentran dos botones. El primero, con la leyenda **“Abrir Archivo”**. Al dar clic sobre este desplegará el explorador de archivos desde el cual podrá seleccionarse un archivo de texto para que su contenido sea visualizado dentro del programa. Junto al botón anterior se encuentra el de **“Limpiar texto”**. Al presionar este botón se borrará el contenido del campo de texto.



La siguiente sección corresponde a la de **“Estructura de Listas”** y se ubica en al lado derecho de la sección anterior**.** Esta se compone de un espacio para desplegar las listas de tokens que el programa vaya generando, dos botones para interactuar con el texto ingresado o generar el listado que será desplegado, y otro botón para generar el archivo HTML, con las listas de tokens tabuladas. Estos botones corresponden a lo que contienen las leyendas **“Tokenizar”** y **“Traducir”** en la parte inferior del campo de listado, y al botón **“Generar HTML”** al costado derecho de este.



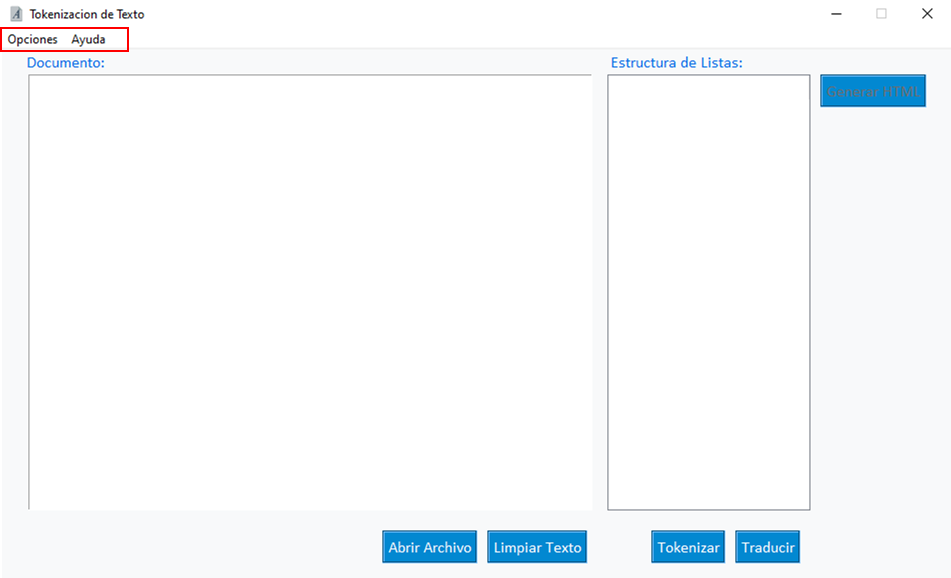
El botón de **“Tokenizar”** se encarga de realizar la clasificación correspondiente sobre cada uno de los elementos del texto que haya sido ingresado. Luego de esto, las listas con los artículos, pronombres, preposiciones, verbos, números y demás palabras serán desplegadas en el campo de listado de la sección izquierda de la pantalla.

El botón de **“Traducir”** realiza una clasificación similar al botón anterior, pero en este caso los tokens serán desplegados en inglés y las categorías a utilizar son solamente las listas con los artículos, pronombres, preposiciones y verbos. De igual manera estas palabras serán desplegadas en el campo de listado de la sección izquierda de la pantalla.

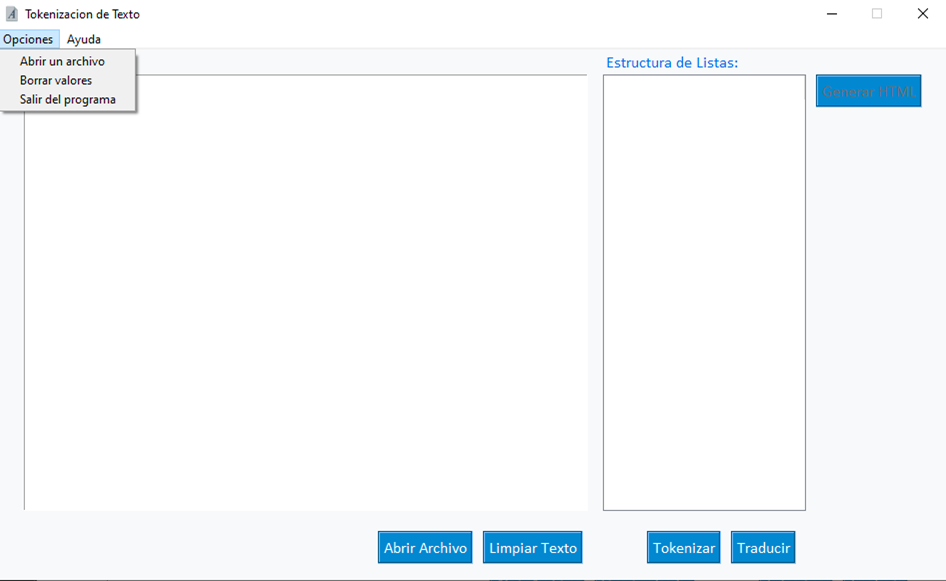
Finalmente, el botón de **“Generar HTML”** se encarga de iniciar el proceso de construcción y guardado del archivo HTML que contiene una clasificación tabulada de las palabras desplegadas en los listados de tokens de la sección izquierda de la pantalla. Este archivo contiene el texto ingresado dentro del programa, seguido de una tabla con todas las clasificaciones y en sus columnas las palabras que pertenecen a cada una de estas. Debajo de esta tabla se incluye también otra con las clasificaciones válidas para los tokens que puedan ser traducidos a inglés, los cuales, tal como se indicó para el botón **“Traducir”,** serían solamente cuatro categorías.

* **Menú de Opciones Superior**

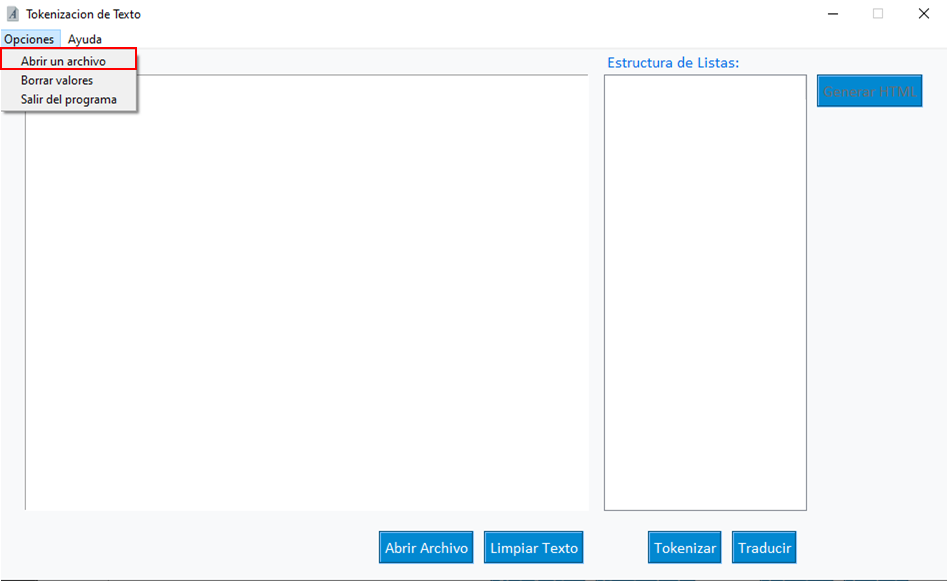
En la parte superior de la ventana principal puede observarse una barra de menú con dos opciones principales: **“Opciones”** y **“Ayuda”**.

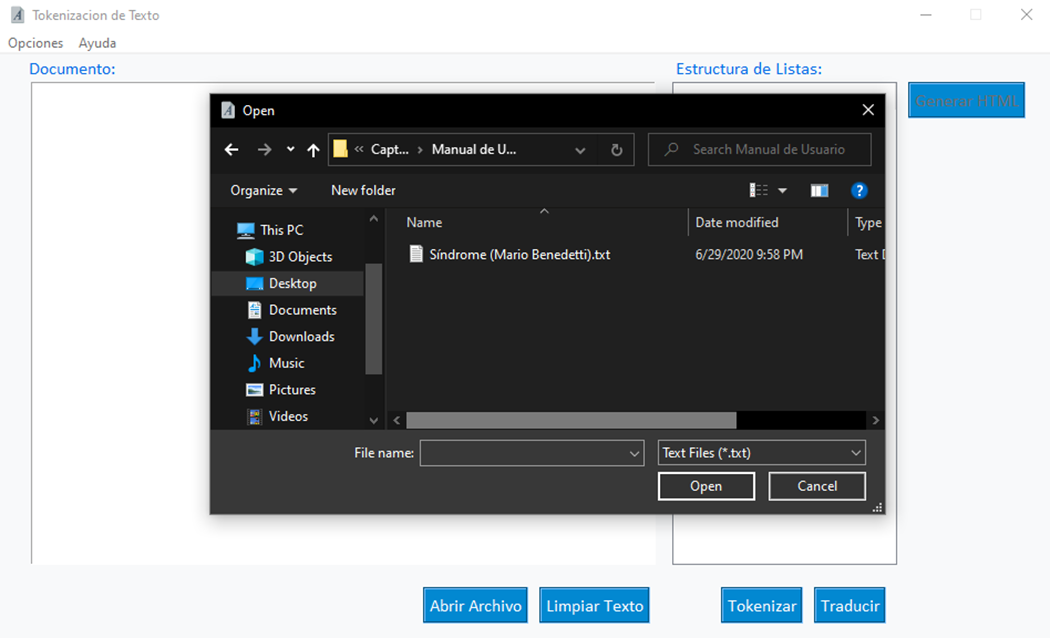


Cada una de las opciones mencionadas corresponde a su vez a un menú desplegable. Al hacer clic sobre la opción de **“Opciones”**, desplegarán algunas de las tareas que puede realizar el programa.

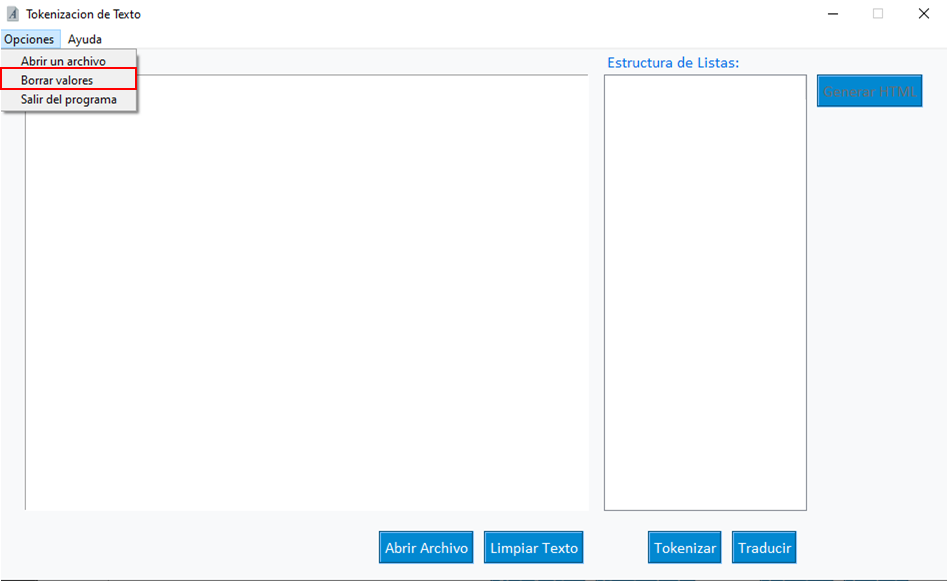


La primera de estas corresponde a **“Abrir un archivo”**, al hacer clic sobre esta se abrirá la ventana del explorador de archivos, desde la cual es posible seleccionar un archivo de texto o con la extensión “.txt” para que su contenido sea extraído por el programa.

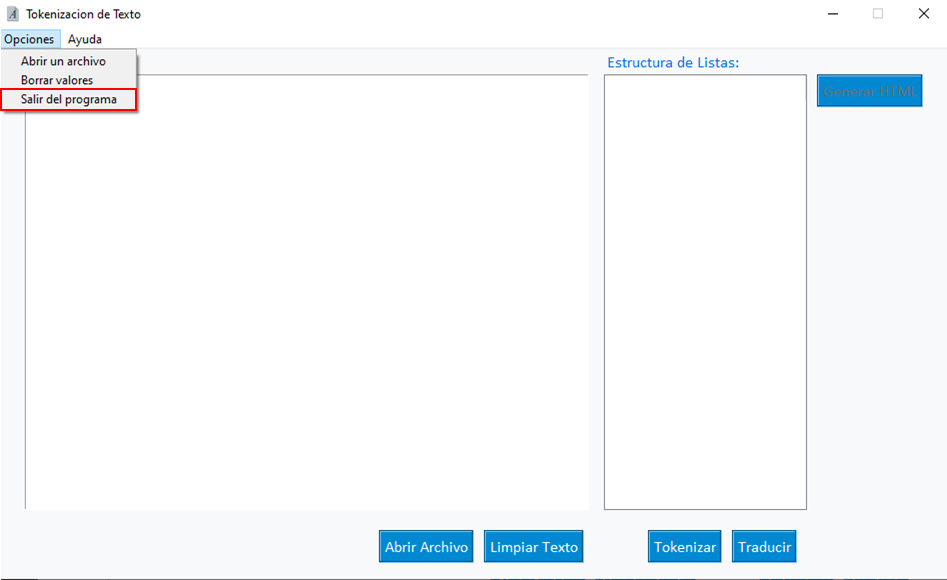




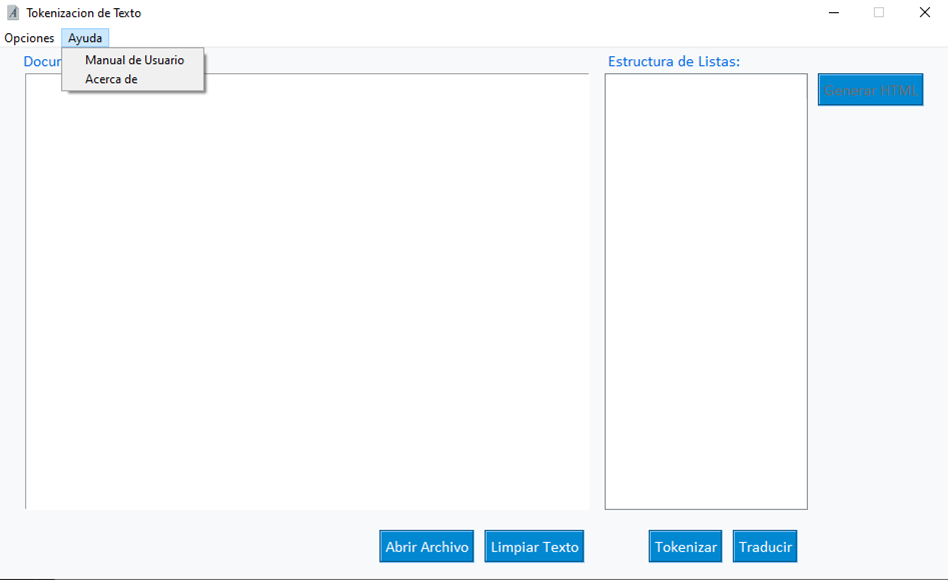
Desde la segunda opción del primer menú desplegable, **“Borrar valores”**, es posible borrar los valores que hayan sido ingresados dentro del programa. en este caso estos valores pueden ser texto que el usuario haya ingresado, texto que se haya leído desde un archivo o incluso los resultados de traducir y tokenizar alguna de las opciones de texto mencionadas.



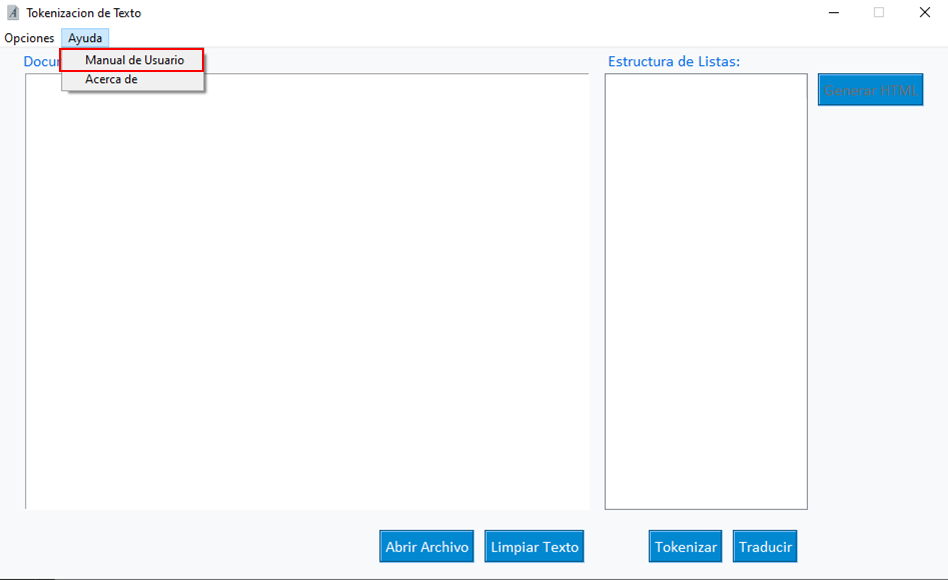
Desde la tercera opción del primer menú desplegable, **“Salir del programa”**, es posible cerrar el programa o detener su ejecución.

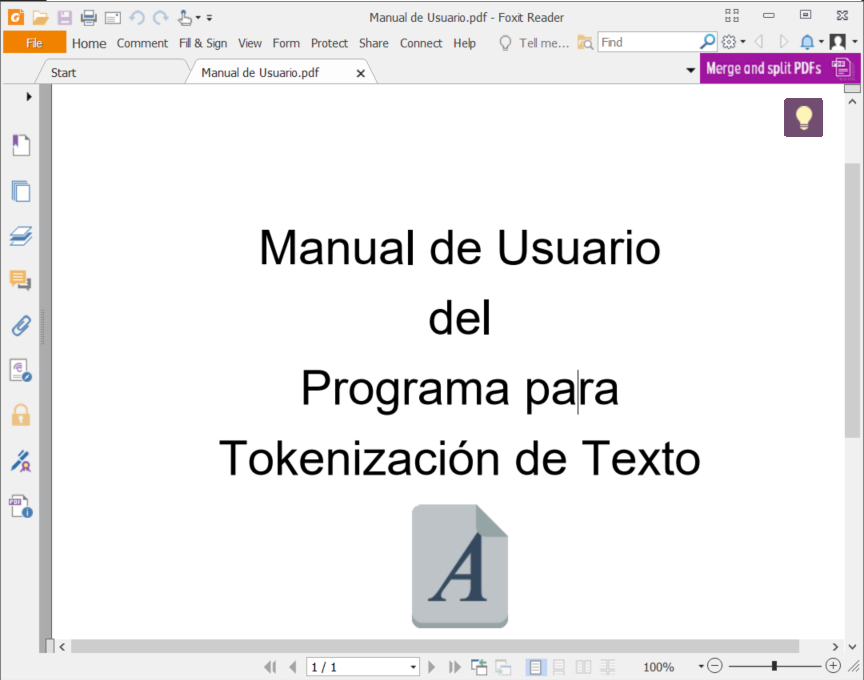


Al hacer clic sobre la opción de **“Ayuda”**, desplegarán algunas de las tareas que puede realizar el programa.

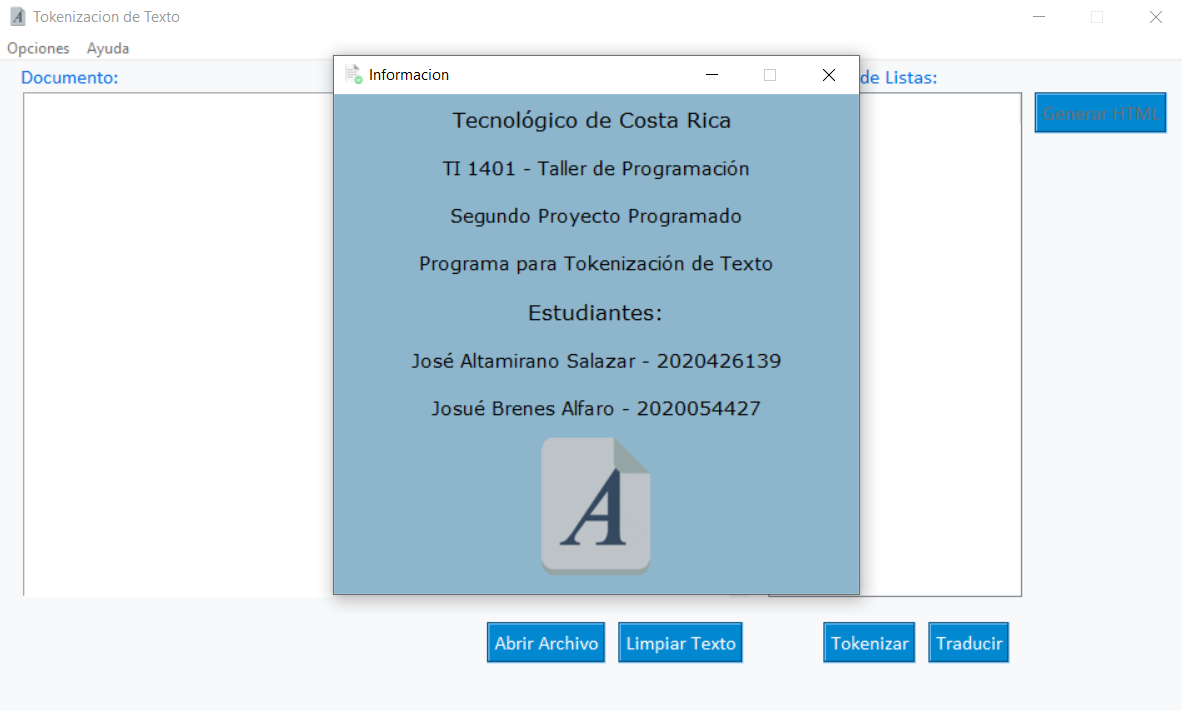


Si se selecciona la primera opción de ese submenú, **“Manual de Usuario”**, el programa le abrirá el documento que contiene el este manual de usuario Esta opción debe ser utilizada en caso de que se requieran consultar los pasos necesarios para algún proceso o tarea del programa.





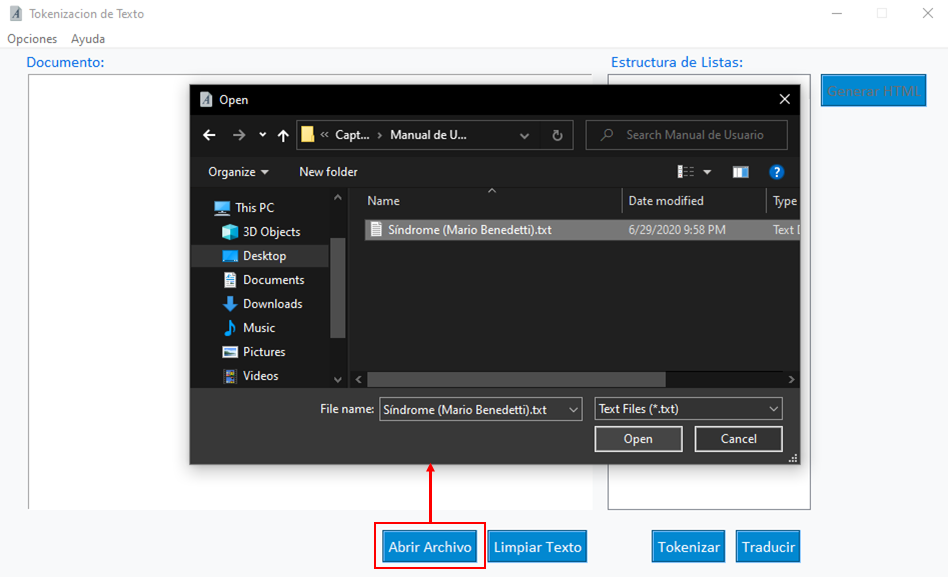
Si se selecciona la segunda opción de ese submenú, **“Acerca de”**, el programa le abrirá otra ventana que contiene información sobre el equipo que desarrolló el software.

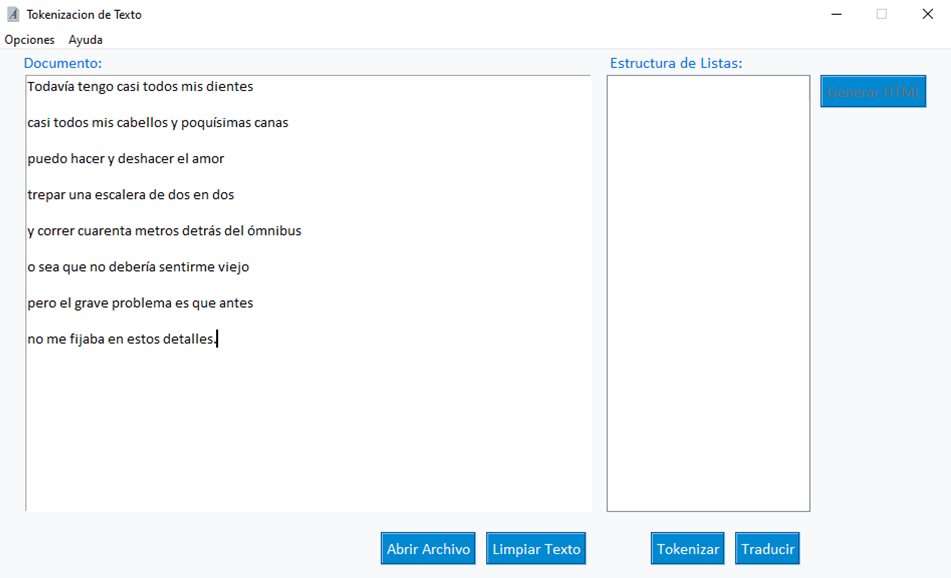
****

* **Tokenizar un texto**

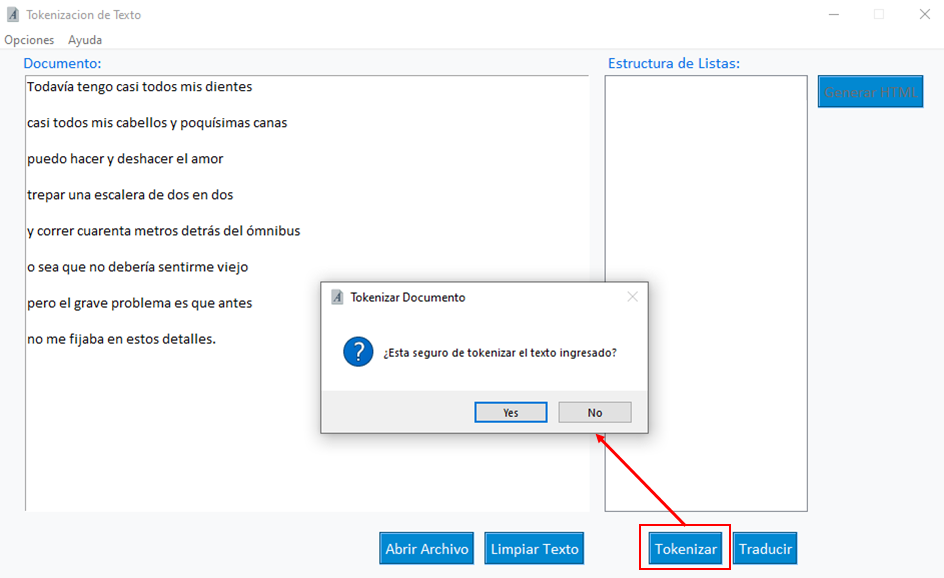
El proceso para tokenizar un texto es muy sencillo, y consta de solamente 4 pasos, los cuales se describen a continuación.

1. Ingrese texto dentro del campo dispuesto para este propósito. Para esto puede dar clic sobre el cuadro ubicado a la izquierda de la pantalla y digitarlo de forma manual o también puede seleccionar un archivo de texto. Para seleccionarlo haga clic sobre la opción del menú superior: **Opciones-->Abrir un archivo** o presione el botón **Abrir archivo** en la parte inferior de la ventana. Al seleccionar un archivo y presionar el botón **“Abrir”**, el texto contenido en este se mostrará en la ventana del programa.

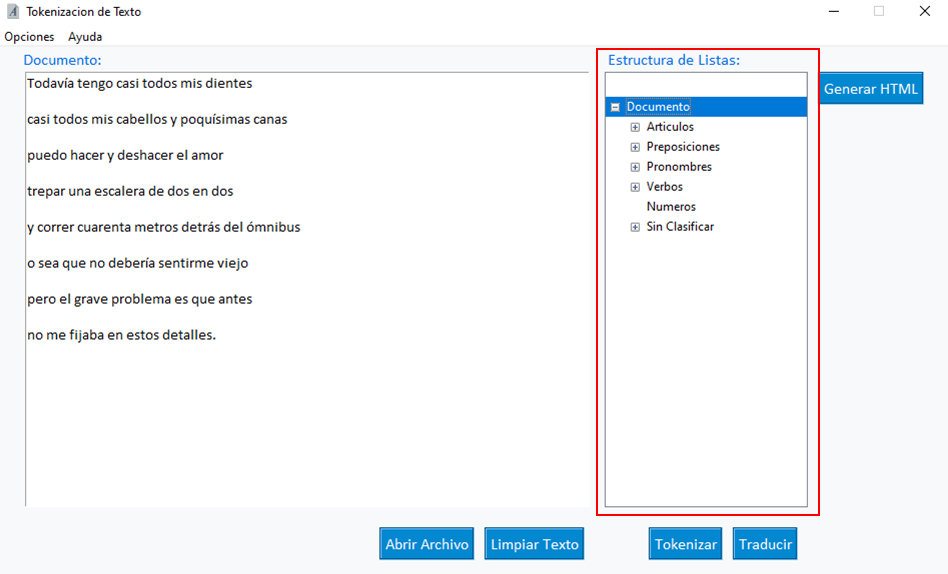




1. En caso de que requiera editar el texto extraído del archivo, puede hacerlo en este momento. Ahora bien, para poder clasificar cada una de las palabras contenidas en el texto, deberá dar clic sobre el botón **“Tokenizar”**. Al hacer esto, se le mostrará una ventana para confirmar la tarea de tokenización del texto ingresado.

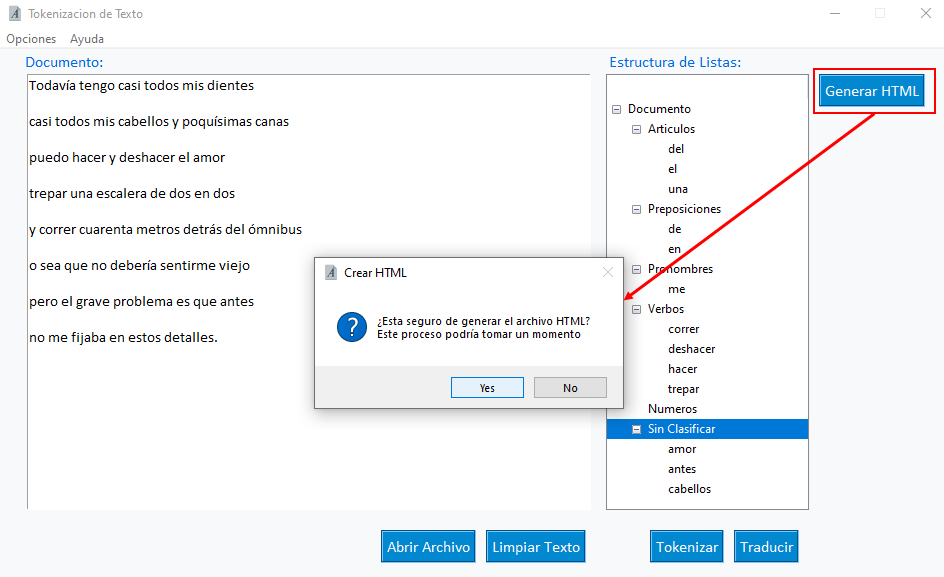


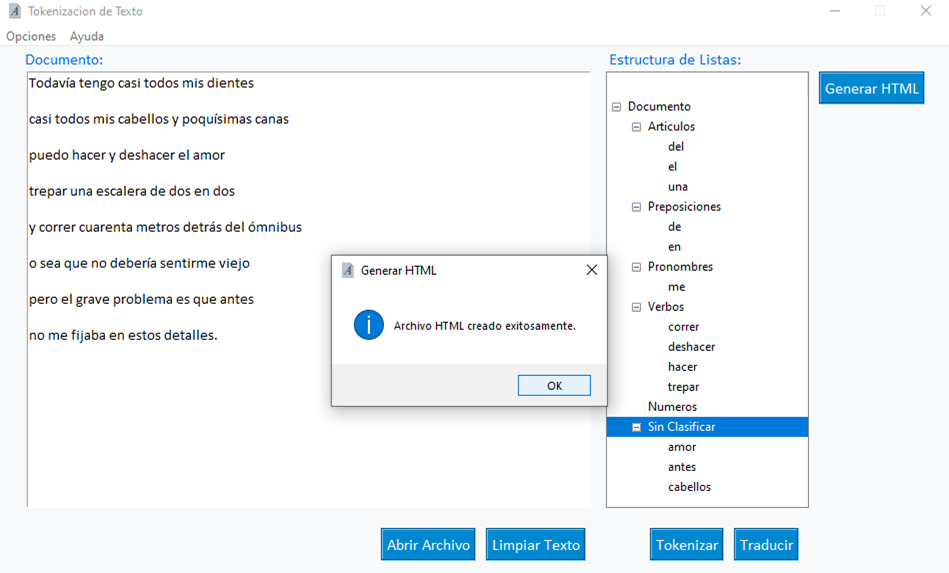
Si presiona que si desea continuar, se cargará el resultado de la operación el campo de listado de la sección izquierda de la ventana.



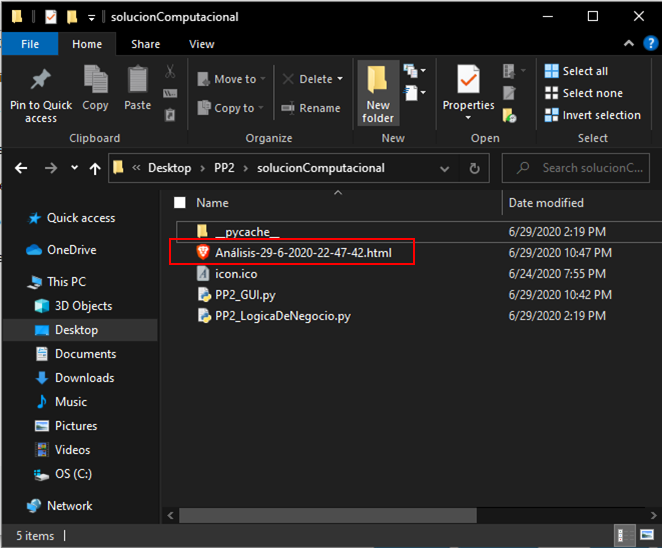
Las opciones del listado pueden ser desplegadas para visualizar todos los elementos contenidos en cada categoría

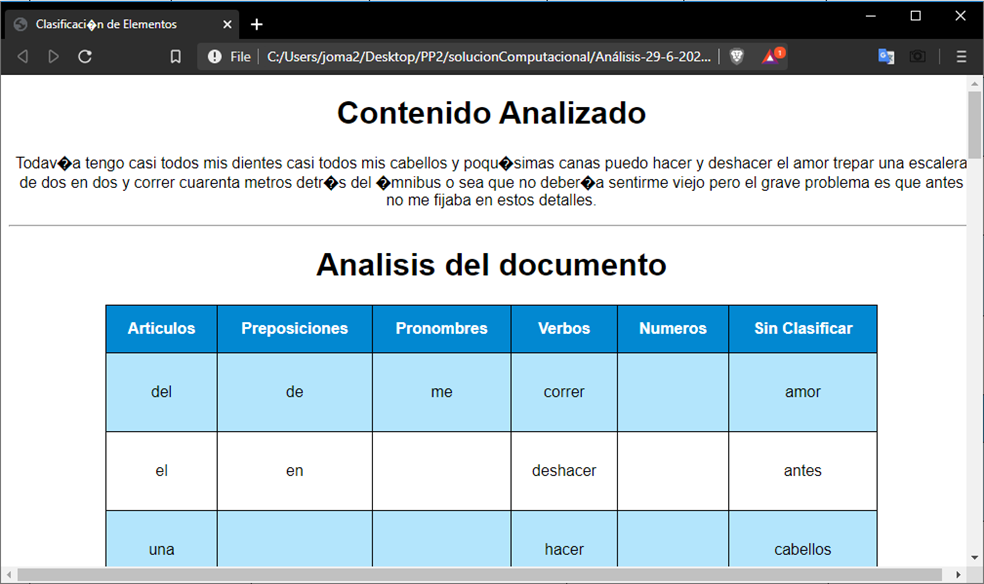
1. Finalmente, para guardar el proceso de la tokenización generado hasta este punto, podrá dar clic sobre el botón **“Generar HTML”**, el cual hasta este punto se ha mantenido deshabilitado. Al dar clic sobre este se mostrar otra ventana para confirmar que se desea generar el archivo correspondiente, y si se si acepta la operación luego de unos segundos me le indicará que el archivo se ha generado de forma correcta.





1. Para poder visualizar el archivo generado, diríjase a la carpeta en donde se encuentra almacenado el archivo ejecutable del programa, y allí podrá encontrar el archivo correspondiente a clasificación recién generada., al dar clic sobre éste el mismo se abrirá en el navegador predeterminado de su equipo, y podrá visualizar la clasificación de los tokens tabulada. El archivo se encuentra guardado con un nombre que incluye la fecha y la hora en fue generado.

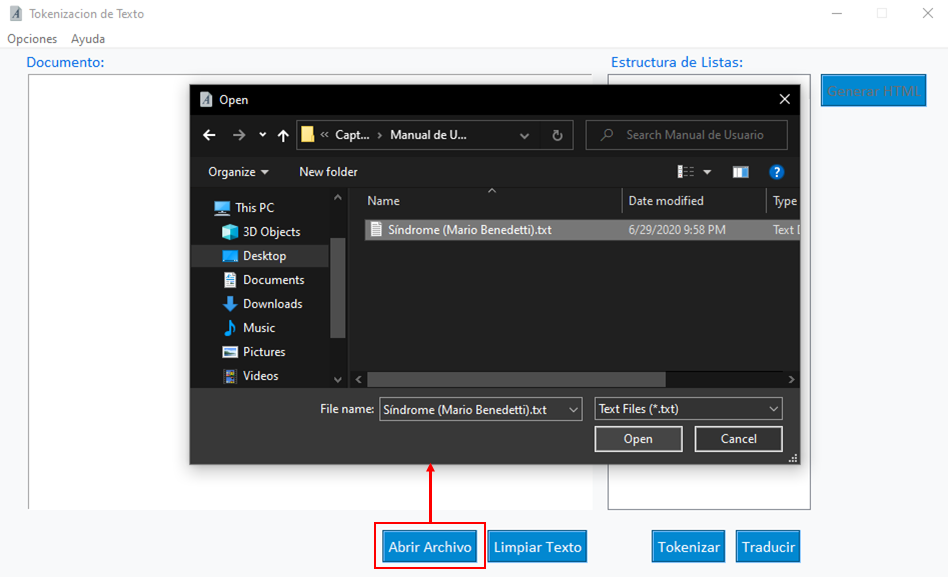


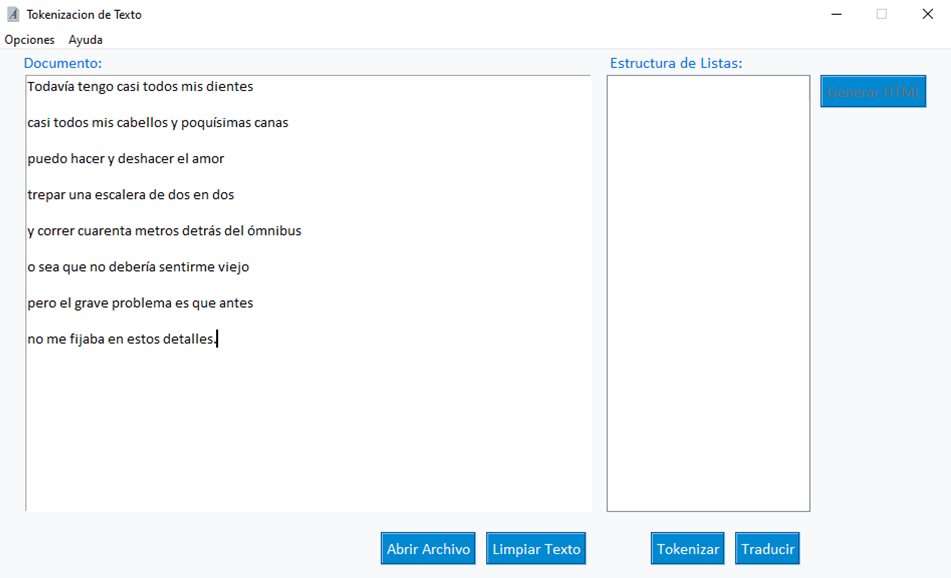


* **Visualizar la traducción de un texto**

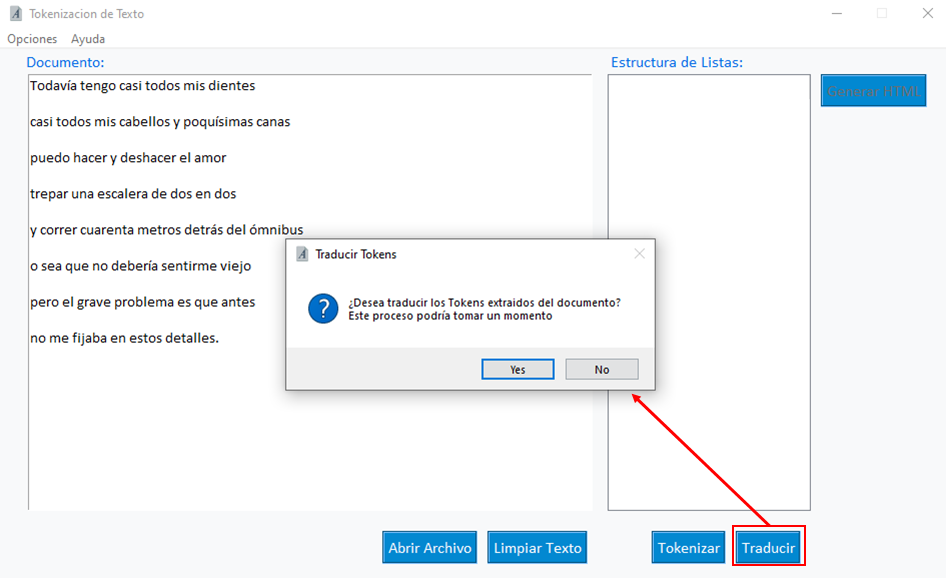
En caso de que se requiera verificar o visualizar la traducción de los tokens dentro del programa, debe seguirse un proceso similar al anterior. De igual manera, los pasos necesarios para completarlo se describen a continuación.

1. Ingrese texto dentro del campo dispuesto para este propósito. Para esto puede dar clic sobre el cuadro ubicado a la izquierda de la pantalla y digitarlo de forma manual o también puede seleccionar un archivo de texto. Para seleccionarlo haga clic sobre la opción del menú superior: **Opciones-->Abrir un archivo** o presione el botón **Abrir archivo** en la parte inferior de la ventana. Al seleccionar un archivo y presionar el botón **“Abrir”**, el texto contenido en este se mostrará en la ventana del programa.

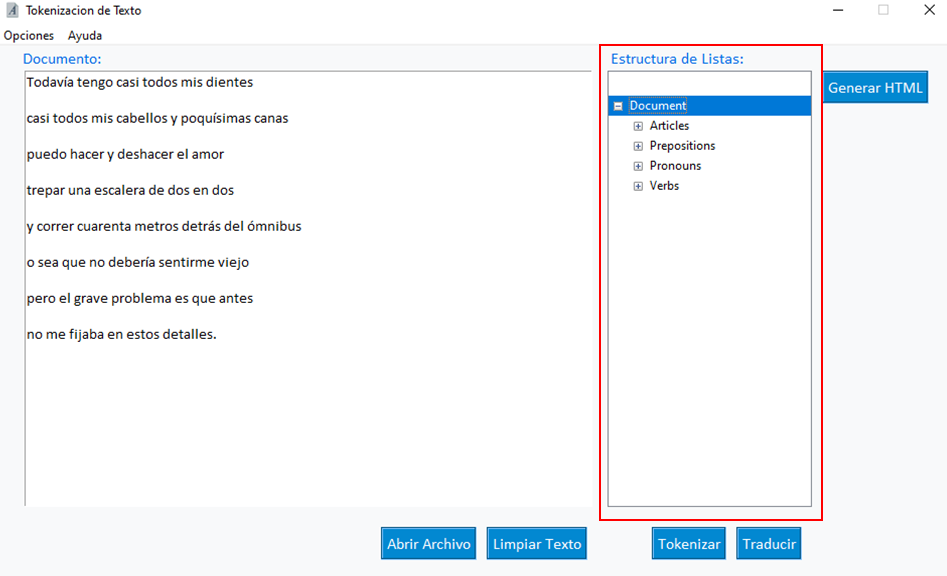




1. En caso de que requiera editar el texto extraído del archivo, puede hacerlo en este momento. Ahora bien, para poder clasificar cada una de las palabras contenidas en el texto, deberá dar clic sobre el botón **“Traducir”**. Al hacer esto, se le mostrará una ventana para confirmar la tarea de traducción del texto ingresado.



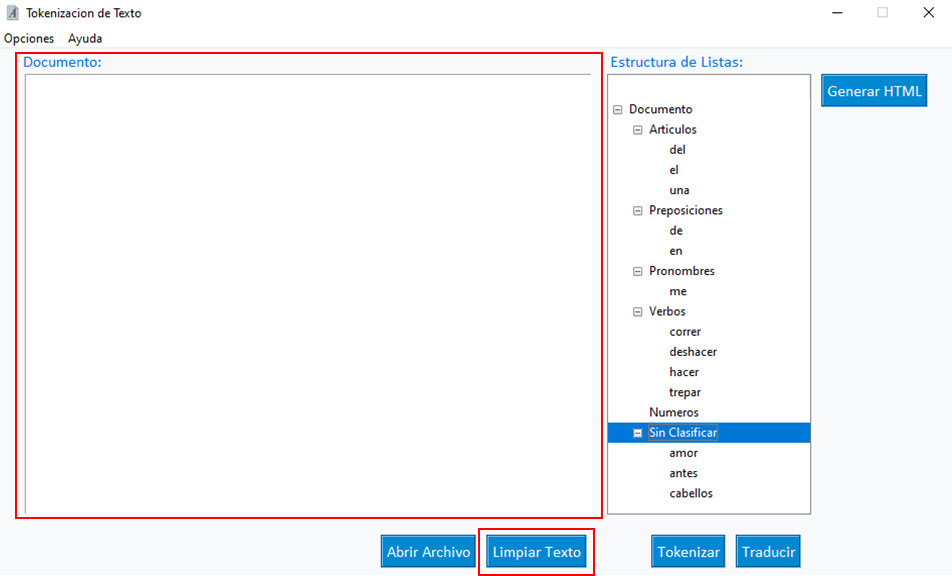
Si presiona que, si desea continuar, se cargará el resultado de la operación el campo de listado de la sección izquierda de la ventana.



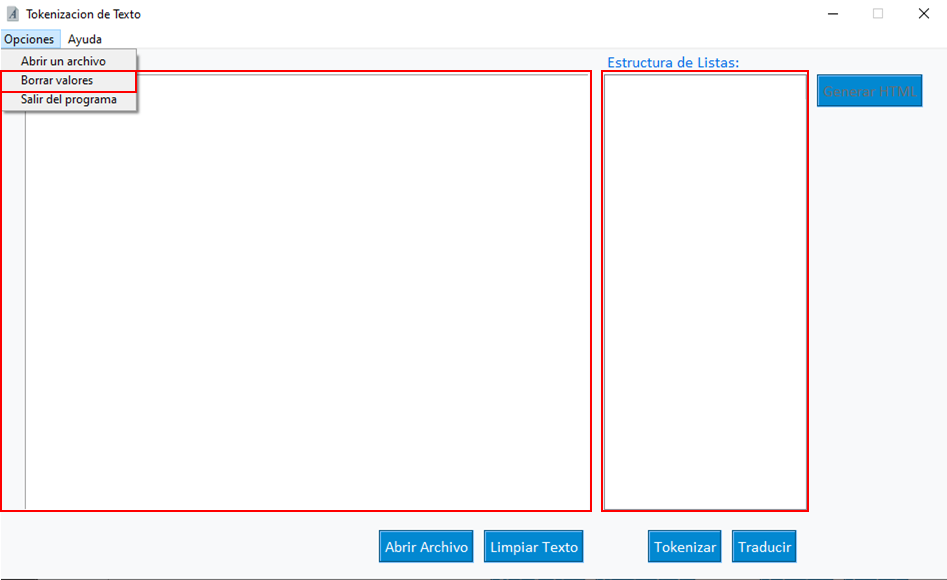
Las opciones del listado pueden ser desplegadas para visualizar todos los elementos contenidos en cada categoría.

* **Borrar los valores ingresados**

En caso de que se requiera eliminar los valores contenidos dentro del programa, se dispone de dos opciones para ejecutar los procesos relacionados a esta tarea. Si solamente es necesario eliminar el texto ingresado en la sección de documento, basta con presionar el botón de **Limpiar texto** de la parte inferior de la ventana. Tenga en cuenta que esta función solamente elimina el texto contenido dentro de la sección mencionada y no así la clasificación realizada en caso de que el texto haya sido tokenizado.

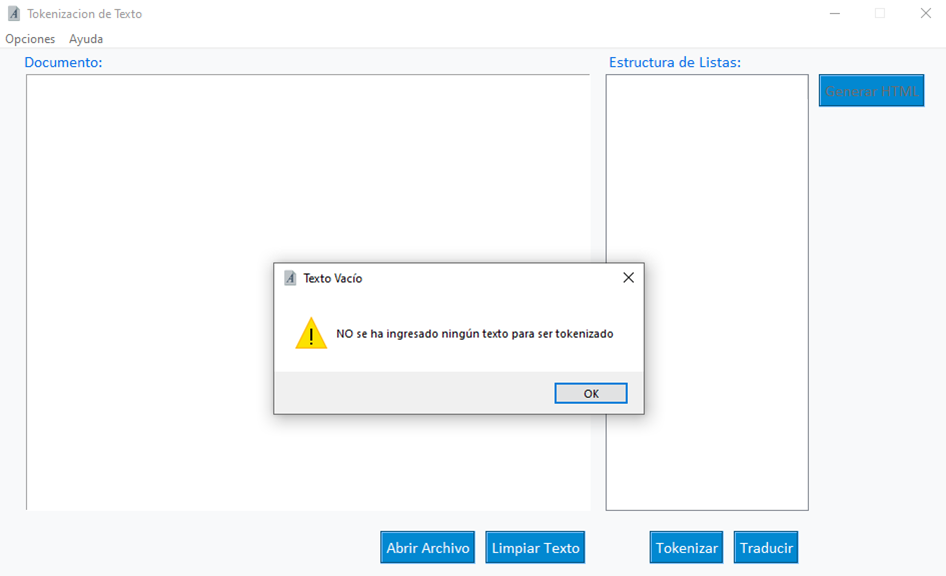


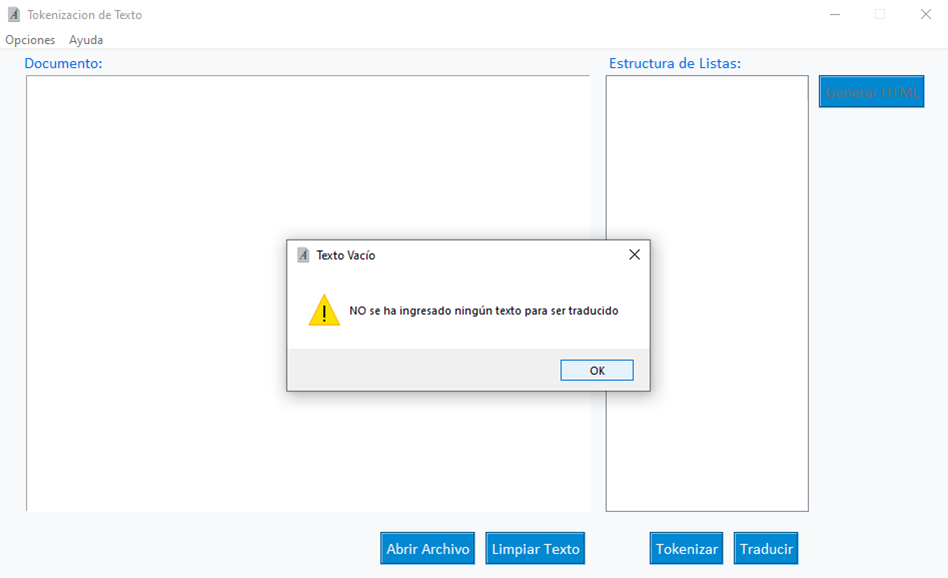
Ahora bien, en caso de que se requiera eliminar por completo todos los valores contenidos dentro del programa en la sesión activa, basta con seleccionar la opción de **Borrar valores,** disponible en el menú superior de la ventana. Esta opción eliminará tanto el texto ingresado como la clasificación generada por el programa.



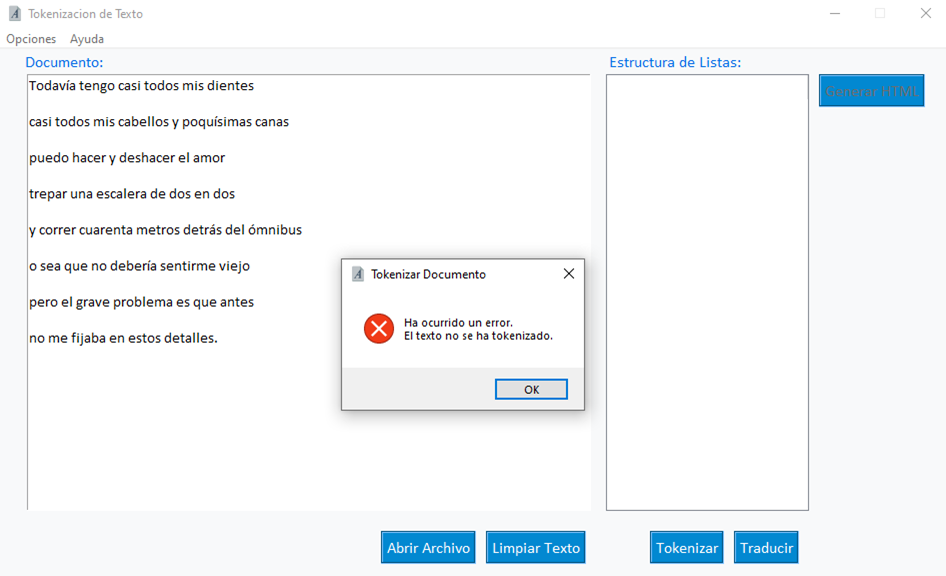
* **Posibles errores dentro del programa.**

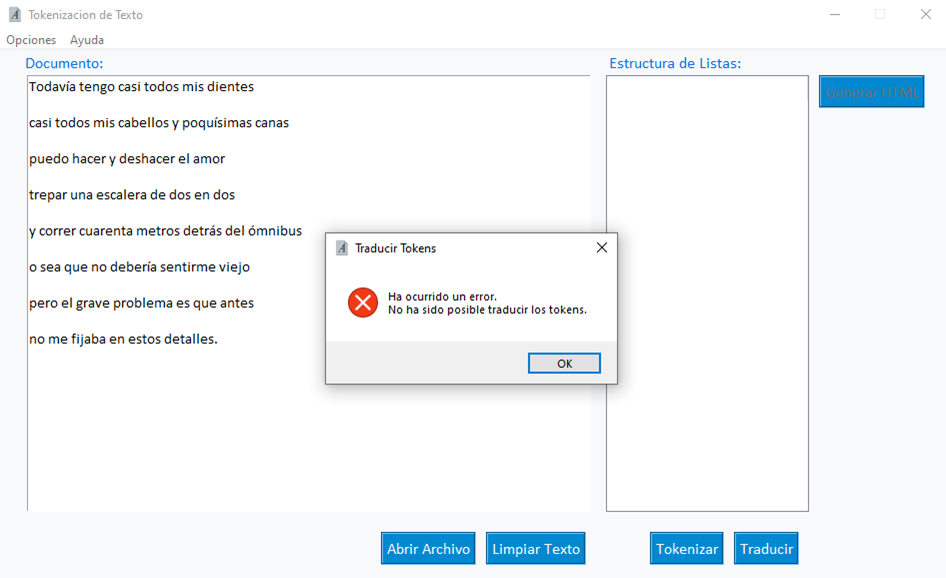
En caso de realizar una acción indebida o que al momento de solicitar la realización de alguna tarea no se hayan incluido toda la información necesaria, el programa le indicará mostrando mensaje de alerta. A continuación, se incluyen ejemplos de los mensajes desplegados por el programa en caso de que no se haya ingresado texto tanto para la tokenización de un texto como para su traducción.





En caso de que en alguno de los procesos antes mencionado, o incluso al intentar generar el archivo HTML no haya sido completado de forma correcta, el programa también le indicará al usuario cuando se produzca un error de este tipo. Ejemplo de estas salidas se incluyen las siguientes imágenes.





# Pruebas de Funcionalidad

En esta sección detallan cada una las funcionalidades implementadas dentro del programa de “Tokenización”. Para cada opción se especifican las posibles entradas o valores que se requiera sean ingresados, las restricciones que rigen sobre cada una de estas entradas, el manejo de errores en caso valores no permitidos o no válidos y las salidas o valores retornados por cada una de las funciones utilizadas.

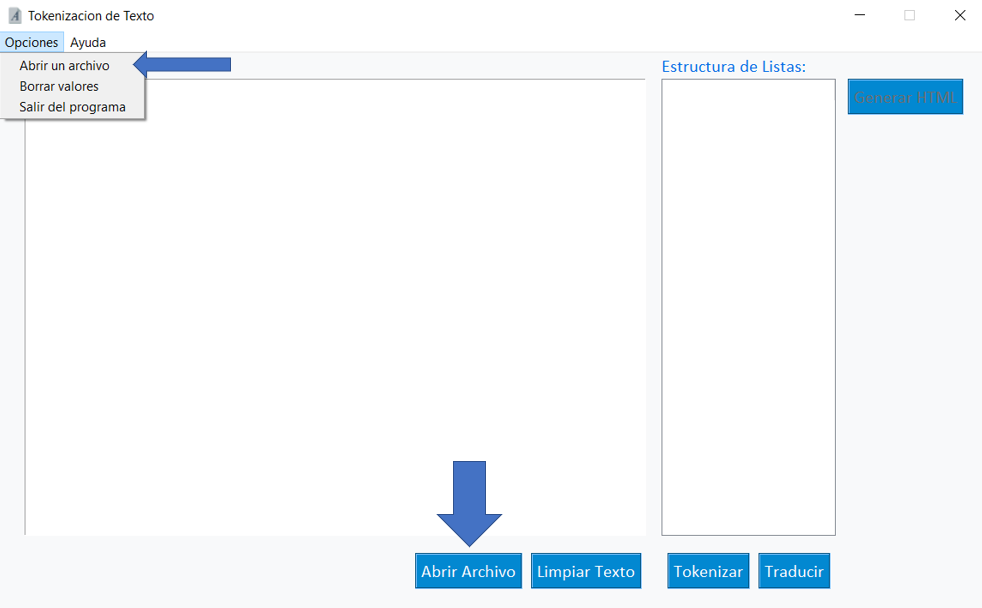
Cada funcionalidad será descrita de acuerdo con la distribución propuesta dentro de la ventana del software. De esta manera podrán estructurarse las descripciones de las diferentes funcionalidades, al hacer énfasis sobre el flujo normal, o acertado y los casos en los que salte una ventana emergente.

## Ventana principal

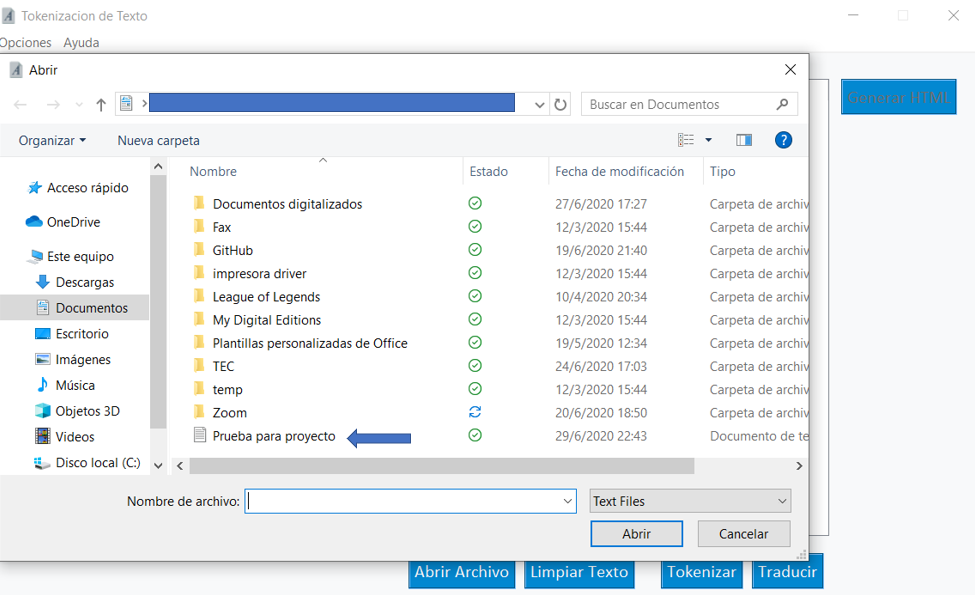
En la ventana principal se encuentra un cuadro de texto con título de **“Documento”**, también un cuadro que en su contenido habrá un árbol de palabras según su clasificación, en el lado superior hay un pequeño menú con dos opciones, y cinco botones diferentes. Se presentará la funcionalidad de la ventana a continuación.



Para tener información dentro de la ventana principal se tiene la opción de abrir un documento .txt que esté creado en la computadora. La opción de abrir el documento se puede hacer de dos formas: desde la zona inferior del cuadro de texto al dar clic sobre el botón de **“Abrir Archivo”** o en el menú superior del cuadro de texto en la pestaña “**Opciones”** trae la opción de **“Abrir un archivo”**. También se puede digitar directamente en el cuadro de texto.

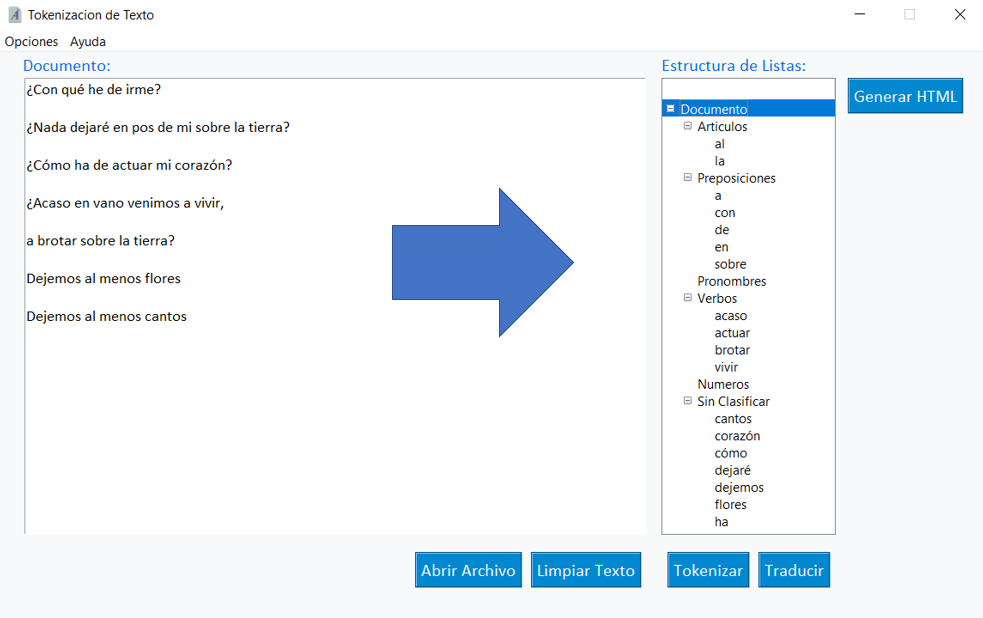


Al tocar el botón de **“Abrir Archivo”** o **“Abrir un archivo”** (Menú superior) se desplegará una ventana en la cual se puede buscar el documento que desee abrir en el cuadro de texto. Solo aparecerán archivos .txt ya que son los únicos que pueden ser ingresados en el cuadro de texto.

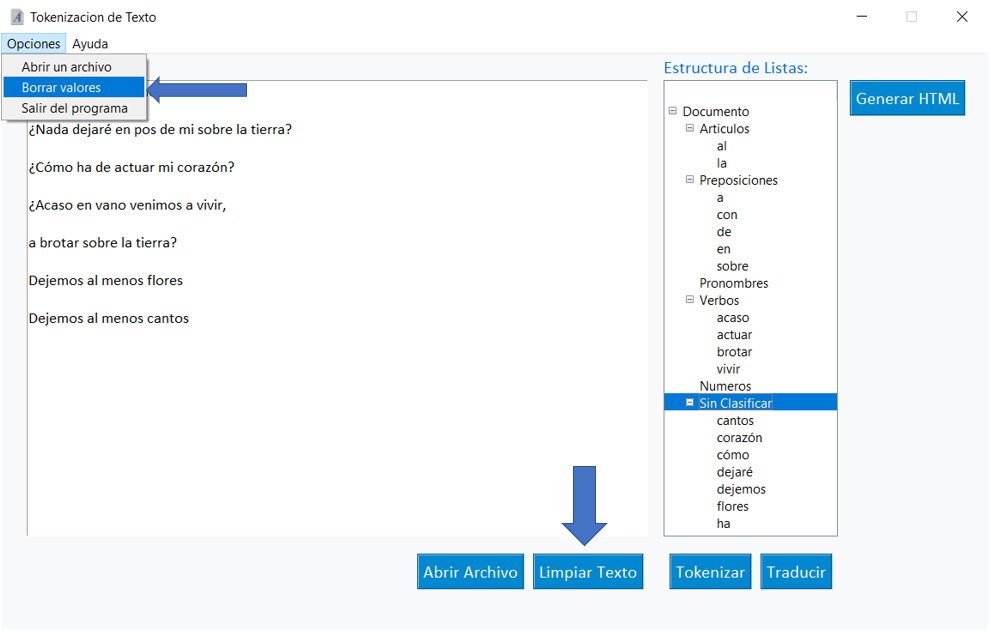




Al tener información el en cuadro de texto se puede tokenizar (separar por palabras) clickeando el botón **“Tokenizar”**. Este separará las palabras en Artículos, Preposiciones, Pronombres, Verbos, Números y los que no se clasifiquen. Estos se presentarán mediante listas en el cuadro con encabezado **“Estructura de Listas”**.

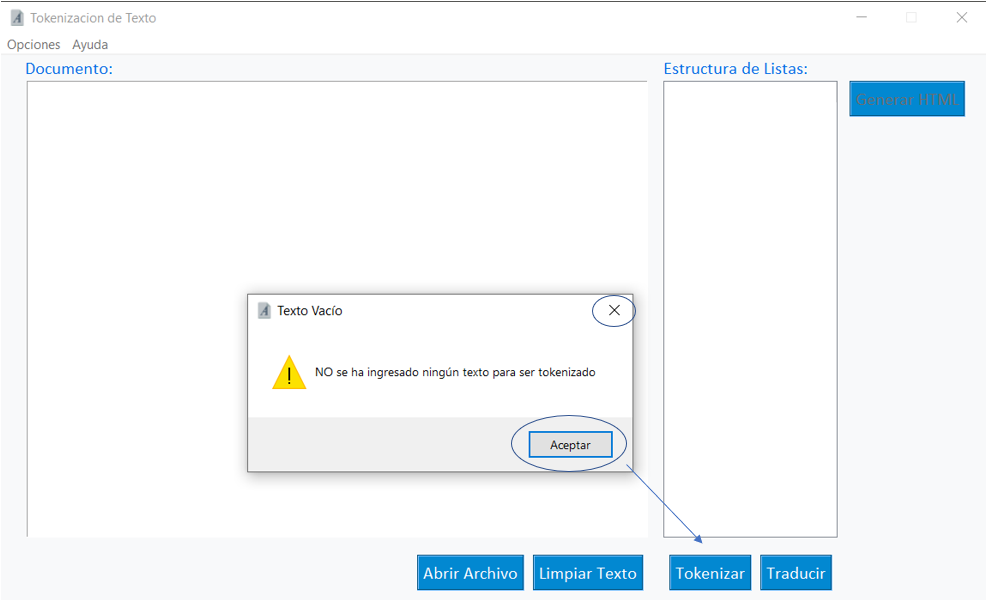


Si se quiere eliminar los elementos que se encuentran en el cuadro de texto hay dos opciones, una sería clickear el botón **“Limpiar texto”** y se eliminará únicamente lo que haya en el cuadro texto, la otra opción es la el menú superior al clickear la opción **“Opciones”**, se desplegará un submenú al seleccionar la opción **“Borrar valores”** se eliminarán los elementos del cuadro de texto, pero también con esta opción se eliminan los elementos que estén en las listas de del cuadro **“Estructura de Listas”**

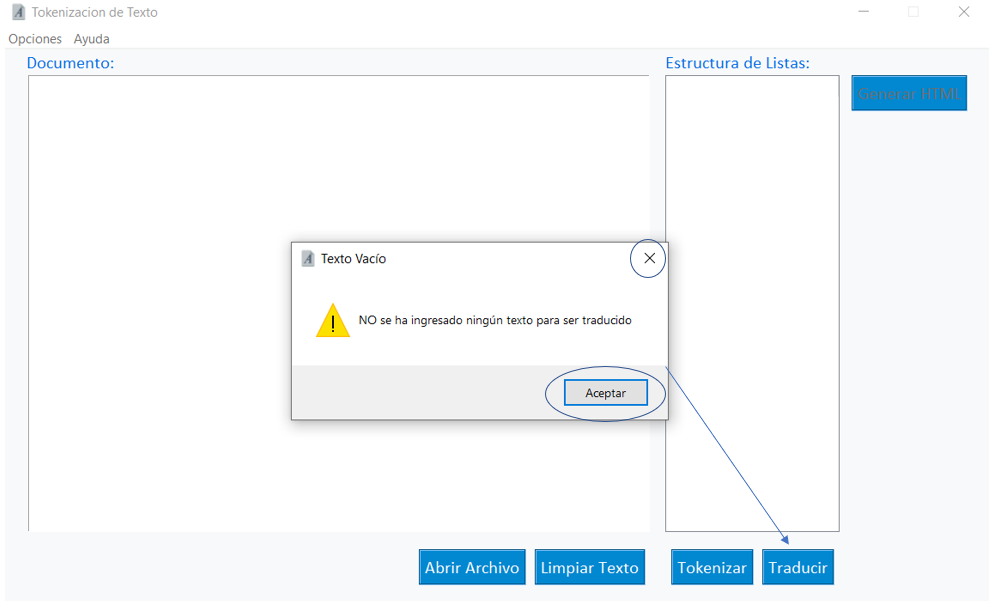




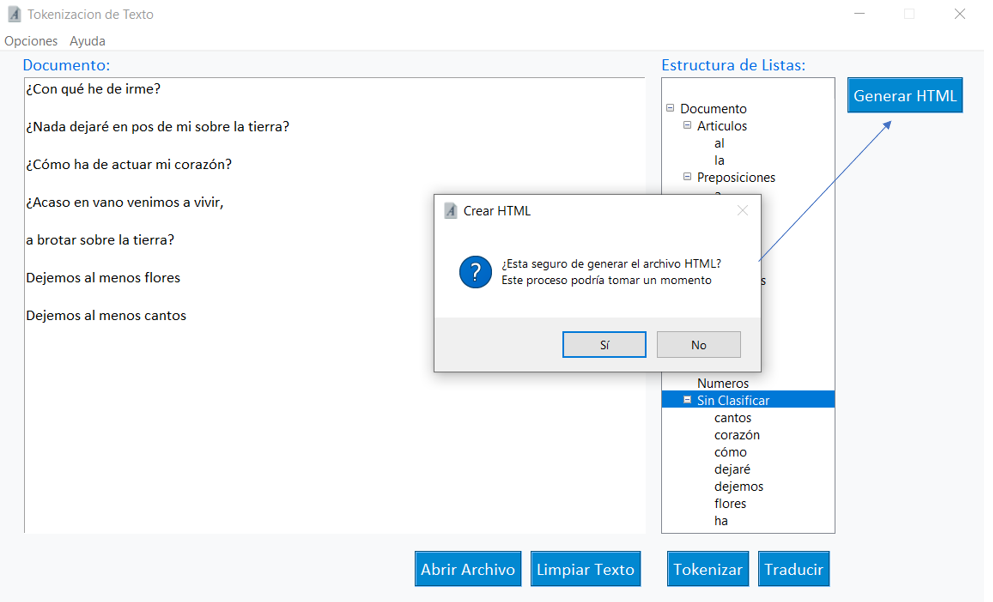
En caso de que no haya información en el cuadro de texto y se presione el botón “**Tokenizar”** saldrá una ventana de advertencia, la cual advierte que no hay texto para ser tokenizado. Se debe presionar **“Aceptar”** o en la **x** superior de la ventana para cerrarla.

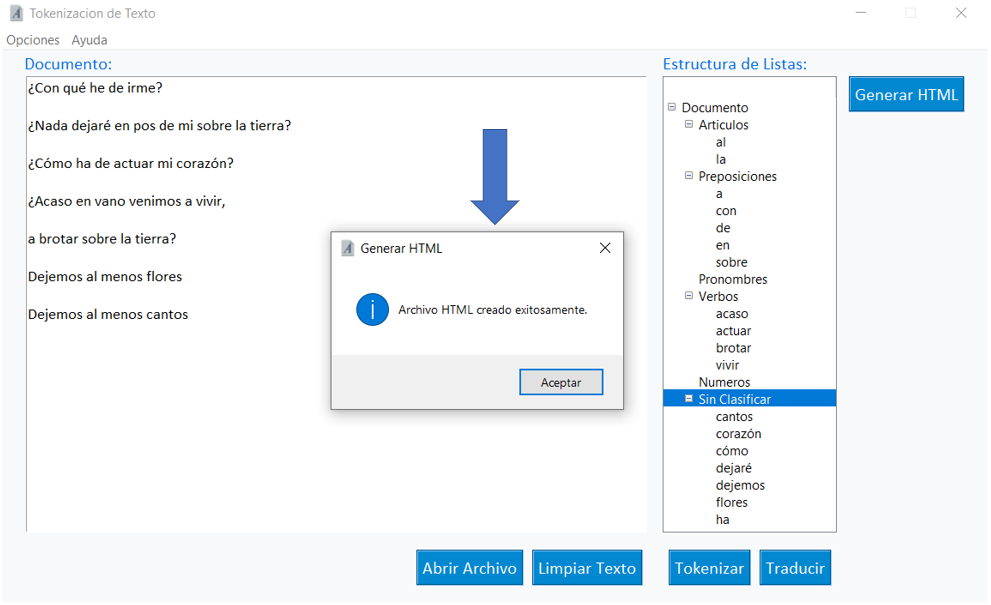


En caso de que no haya información en el cuadro de texto y se presione el botón de **“Traducir”** saldrá una ventana de advertencia, en la cual advierte que no hay texto para ser traducido.

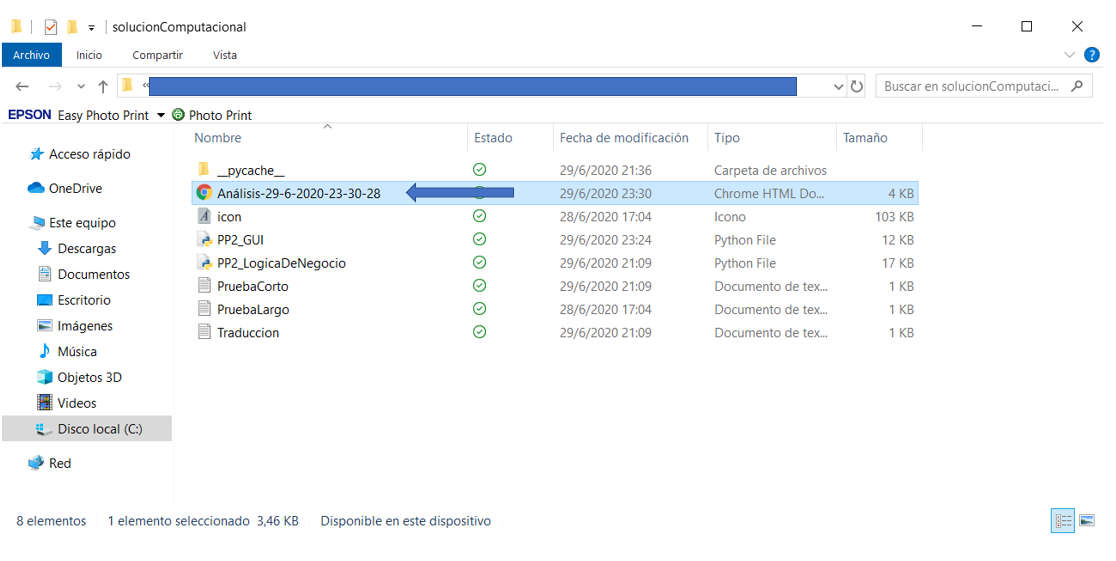


Al tener información en el campo de texto y haber tokenizado esa información con el botón de **“Tokenizar”**, se puede generar un archivo HTML en el cual se crea una tabulación de las listas que aparecen en el campo de **“Estructura de listas”** y además genera otra tabla en la cual traduce los elementos de Artículos, Preposiciones, Pronombres y Verbos. Al darle al botón “sí” de la ventana de pregunta, deberá dar un momento a que se genere el archivo HTML.

 Cuando el archivo se haya creado con éxito aparecerá en pantalla un mensaje que indica que el archivo se ha creado con éxito.



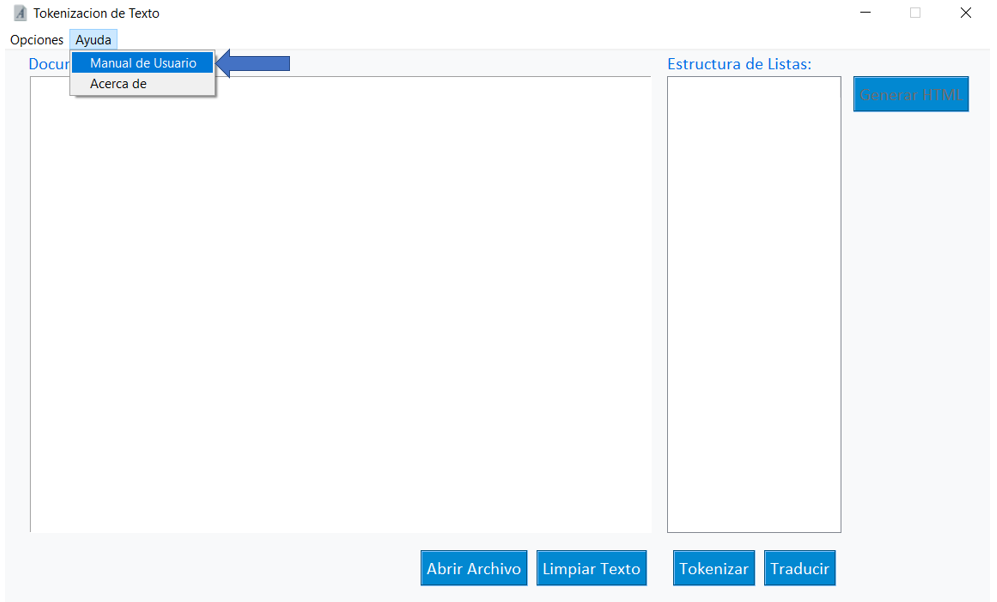
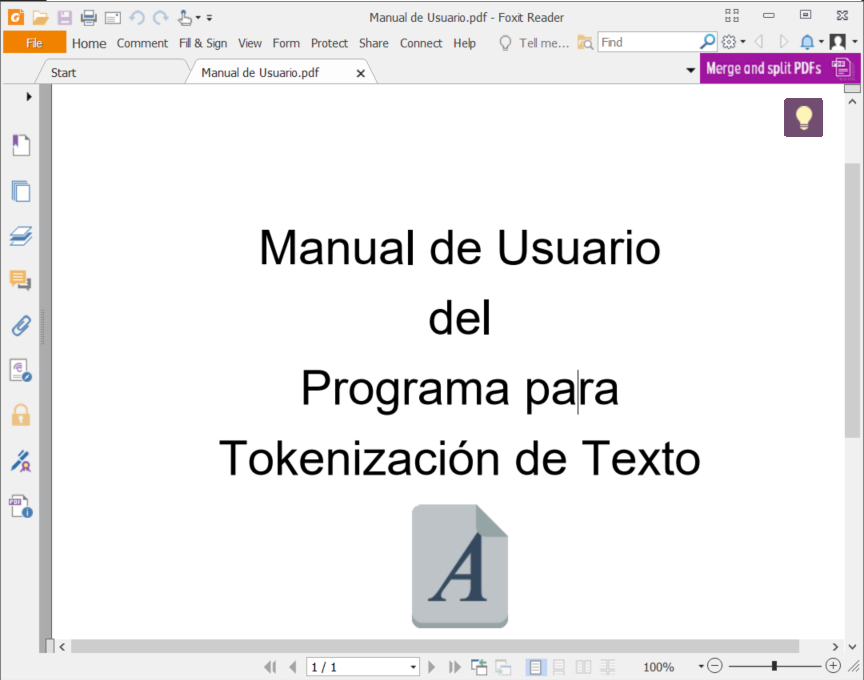
Una vez tengamos el archivo creado con éxito. Para abrir este documento HTML se tiene que dirigir donde se encuentra el archivo ejecutable del programa el cual tendrá la fecha y hora del momento en el que fue creado.



Al abrir este archivo se desplegará en su navegador predeterminado el archivo en el cual se encontrará las tablas resultantes de la tokenización anteriormente hecha.

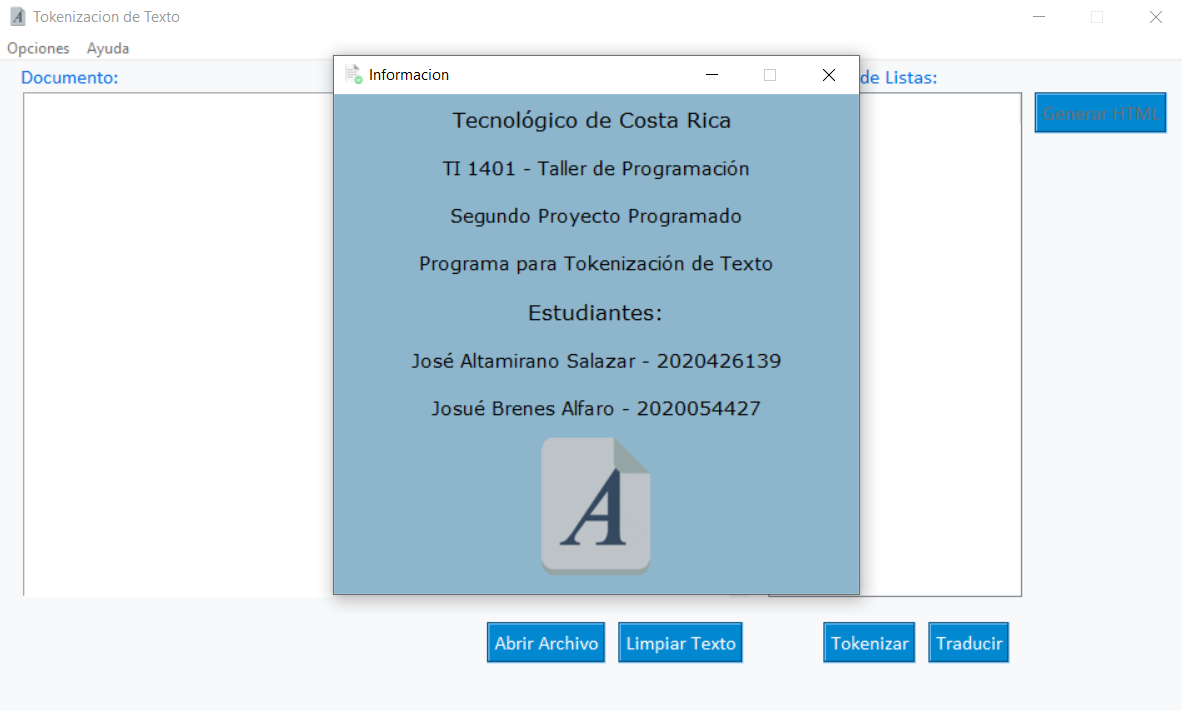


En el submenú de la ventana principal hay dos opciones en las darle a la opción **“Ayuda”** se despliegan dos opciones. En **“Manual de Usuario”** se desplegará el manual de usuario del sistema en el cual aparecerán todas las instrucciones necesarias para usar el sistema de Tokenización de Texto

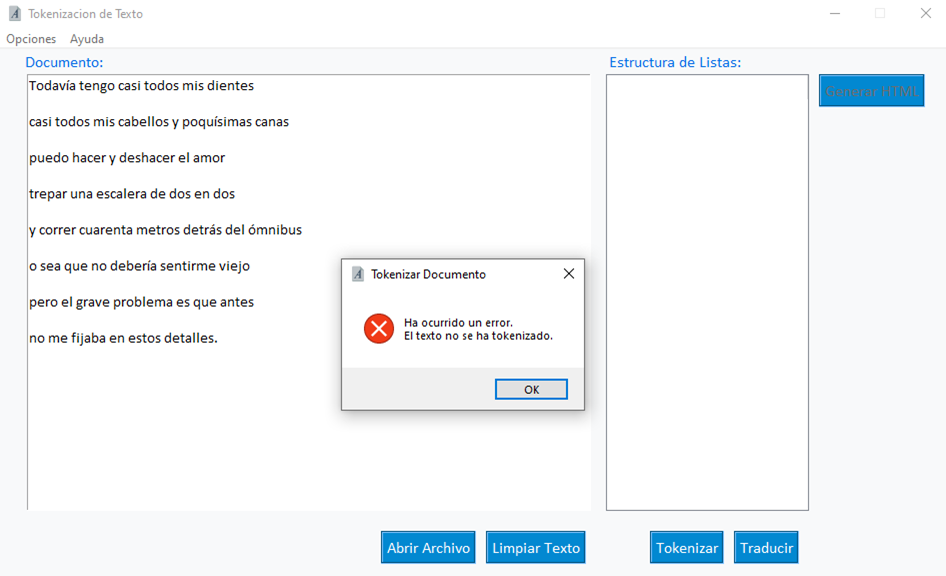


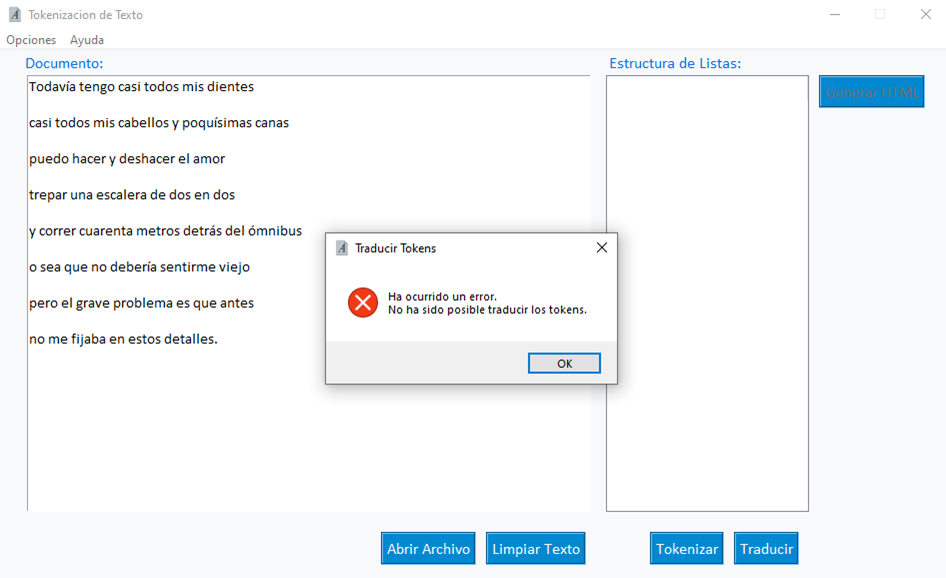
Si se selecciona la segunda opción de ese submenú, **“Acerca de”**, el programa le abrirá otra ventana que contiene información sobre el equipo que desarrolló el software.

****

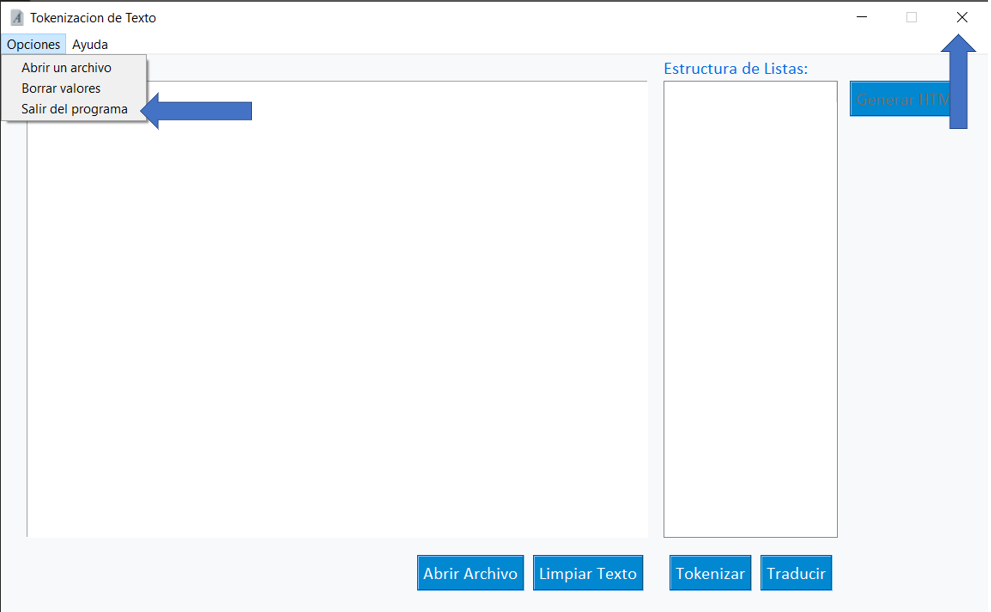
****

En caso de que las funciones de tokenizar o traducir tengan algún error, se presentará en la pantalla una ventana notificándolo. Puede dar click en el botón de **“ok”** o en la **“x”** superior derecha de la ventana pequeña para continuar.





Cerrar el programa se puede hacer de dos maneras, la primera con la **X** de la esquina superior derecha. También en el menú superior en la opción **“Opciones”** se desplegará un submenú, y en la última de esas opciones **“Salir del programa”** cerrará el programa.



# 

# 

# 

# 

# 

# Lecciones Aprendidas

**Josué Brenes Alfaro:**

Durante este proyecto, en lo que a mí respecta, he tenido un aprendizaje hacia la programación muy basto, aún mayor al del primer proyecto. Ya que hubo una variable nueva que fue la creación de una interfaz gráfica, para la cual se tuvo que investigar y hacer muchas pruebas. Gracias a mi compañero Jose Manuel Altamirano, aparte de que es una gran persona, su habilidad de programación es muy grande, y gracias a ello he logrado aprender muchas cosas a la hora de programar. Por ejemplo, en el desarrollo de la lógica en cada función, tener un planteamiento del problema y resolverlo de la manera más limpia y rápida. También algo que se aprende al trabajar en parejas, es que se tiene que ser responsable con las reuniones, aportes y decisiones que se toman reto a reto.

Uno de los aprendizajes descubiertos en este proyecto es trabajar con tkinter para la creación de interfaces gráficas. La utilización de esta librería es de agrado, ya que hay muchas formas de crear una ventana, como poner colores, tipos de fuentes a textos, botones, cuadros de textos, un “tree view” para la creación del árbol con el orden de los tipos de palabras que alberga el cuadro de texto y cada una de estas cosas se pueden modificar a gusto propio.

En un principio se empezó con la utilización de PyQT que es un diseñador de ventanas sin embargo no se tenía conocimiento si se podía utilizar en el proyecto. Dado esto, se decide utilizar la librería tkinter combinada con funciones, lo cual conllevo una investigación de su implementación. Esto se hizo de diversas formas como búsqueda de información en internet, asistiendo a talleres que impartieron los tutores del curso y consultando al tutor del grupo. Después de varias pruebas se obtuvo como resultado una interfaz amigable a la vista y se utiliza de una manera muy sencilla, y con una funcionalidad completa.

Se hizo una división de archivos con las funciones, en las cuales un archivo contiene la “lógica de negocio” en la que se albergan las funciones que interactúan con los datos directos, mientras que el otro archivo contiene el “GUI” en el cual se encuentran las funciones y objetos que interactúan directamente con el usuario como los botones y cuadros de textos.

Otro conocimiento aprendido es la utilización de librerías para traducir palabras en este caso fue de español a inglés, la cual se aplicó en la ventana principal dentro del “tree view”, y en la creación del archivo HTML. Esto fue un buen aprendizaje ya que en Python se implementó una función para crear un archivo HTML que contiene dos tablas, una con las palabras clasificadas y otra con las palabras traducidas a inglés gracias al API traductor de Google.

**José Manuel Altamirano**

Al finalizar este proyecto, han sido muchos los aprendizajes que se obtuvieron. En lo respecta a la codificación, las lecciones son principalmente en relación con el conocimiento adquirido sobre el desarrollo de interfaces gráficas de usuario y el manejo de sus objetos y algunas librerías. Aunque en su mayoría el desarrollo de la interfaz se realizó utilizando la librería Tkinter, durante las primeras etapas del proyecto trabajó también con la librería PyQt. El desarrollo con esta librería y su herramienta de diseño luego fue sustituido por las opciones dentro de la librería Tkinter, sin embargo, su utilización nos proporcionó un mejor entendimiento sobre las diferentes posibilidades dentro del trabajo con objetos gráficos.

Otro punto importante del aprendizaje, en relación con el desarrollo de interfaces gráficas, fue la compresión de la forma en cómo interactúa este elemento del programa con la lógica interna. Por la manera en cómo fue desarrollado el proyecto, integrando la funcionalidad interna del programa junto a interfaz gráfica, fue posible emplear algunas estrategias para facilitar la comunicación entre ambos componentes. El conocimiento y la percepción de este tipo de interacción dentro del programa nos será útil para el desarrollo de soluciones futuras, enfocadas en ofrecer a los usuarios una forma más sencilla de relacionarse con las funciones de los programas.

Para completar este proyecto se han empleado técnicas y estructuras variadas, combinando muchos de los conceptos aprendidos durante las lecciones. Algunos de estos incluyen el trabajo con listas, estructuras iterativas y recursivas, ordenamiento e incluso algunos algoritmos necesarios para la búsqueda de elementos. La utilización de estos conceptos dentro de proyecto dotó de un espacio para ampliar la compresión que se tenía sobre su correcta utilización y funcionamiento. Las diferentes aplicaciones de estos contenidos dentro del proyecto proporcionan una visión más clara sobre este aprendizaje.

A diferencia del proyecto anterior, en este caso fue necesario emplear más funciones y librerías ya creadas o propias del lenguaje Python. Tanto el uso de funciones con propósitos de cálculos o transformación de cadenas de texto, como otras con funcionalidades un poco más complejas, fue necesario un proceso de exploración sobre la forma en que podrían aplicarse a los requerimientos del proyecto. Esto dio como resultado una compresión sobre las diferentes opciones, apoyos y versatilidad que ofrece el lenguaje Python, y las múltiples aplicaciones que pueden generarse utilizando muchas de estas funciones para la solución de diversos problemas.

Al igual que el proyecto anterior el trabajo en equipo también ha sido un factor determinante para este trabajo. El apoyo de mi compañero Josué Brenes ha sido fundamental para poder dar solución al reto planteado por el proyecto. Más aún ahora que nos encontramos en una etapa avanzada del semestre. La posibilidad de coordinar y dividir las tareas de una manera que favorezca al desarrollo del proyecto, y se adapte a las responsabilidades de ambos, ha resultado muy favorable para el desarrollo de una solución que cumpla con las nuestras expectativas, y con los requerimientos planteados.

.

# 

# Bitácora de Trabajo

En esta sección se incluyen los accesos a las bitácoras de trabajo de los integrantes del grupo. Para cada una se indica el nombre de quien la desarrolla y el enlace por el cual puede accederse a las publicaciones realizadas.

Josué Brenes Alfaro

Bitácora: <https://proyecto2bitacora2020054427.blogspot.com/>

José Manuel Altamirano Salazar

Bitácora: <https://2020426159-pp2.blogspot.com/>

# Análisis de Resultados

## Objetivos Alcanzados

* Desarrollar en forma completa la función de tokenizado de texto enunciada como requerimiento dentro del enunciado del proyecto.
* Conocer la forma en cómo se administran los archivos, principalmente de tipo texto plano, a través de las funciones propias del lenguaje de Python.
* Proporcionar la opción para que programa pueda recibir entrada tanto manual como por medio de archivos de texto plano.
* Identificar los elementos que componen un HTML, de manera que fuera posible la persistencia de los datos al generar un archivo de este tipo como salida del programa.
* Incluir las validaciones necesarias para evitar que se den errores dentro de los procesos de tokenización, traducción de tokens o generar un archivo HTML, así como informar al usuario sobre las necesidades de las entradas o posibles errores dentro de cada proceso.
* Adquirir nuevos aprendizajes sobre el uso del lenguaje Python, específicamente sobre el desarrollo de interfaces gráficas como una herramienta para facilitar la interacción de los usuarios con los programas.
* Comprender la forma en cómo interactúan los diferentes objetos o elementos de un programa de interfaz de gráfica entre sí, o con los cálculos o funciones internas de la lógica del programa.
* Explicar las funcionalidades del programa a través de un Manual y la exposición de sus diferentes casos de uso dentro de la documentación del proyecto.

## Objetivos No Alcanzados

Consideramos que todos los objetivos propuestos al inicio del proyecto fueron logrados y no se ha dejado por fuera ninguna funcionalidad del programa.

# Distribución del trabajo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Núm.. De Tarea** | **Nombre De La Tarea/Actividad** | **Persona Asignada** | **Fecha De Inicio** | **Fecha De Fin** |
|
| 1 | Actividades Generales |  |  |  |
| 1.1 | Registro del Equipo de Trabajo | Jose A y Josué B | 18/6/20 | 18/5/20 |
| 1.2 | Lectura y discusión del proyecto | Jose A y Josué B | 18/6/20 | 19/6/20 |
| 1.3 | Creación del Blog | Jose A y Josué B | 18/6/20 | 19/6/20 |
| 1.4 | Investigación para desarrollo de interfaz gráfica | Jose A y Josué B | 19/6/20 | 23/6/20 |
| 1.5 | Investigación sobre librerías adicionales | Jose A y Josué B | 19/6/20 | 23/6/20 |
| 2 | Actividades de Desarrollo de Código |  |  |  |
| 2.1 | Desarrollo de Interfaz gráfica del programa |  |  |  |
| 2.1.1 | Construcción, pruebas y diseño de interfaz | Jose A | 20/6/20 | 24/6/20 |
| 2.1.2 | Opción para seleccionar y abrir un archivo de texto | Josué B | 24/6/20 | 26/6/20 |
| 2.1.3 | Opción para tokenizar | Jose A | 26/6/20 | 29/6/20 |
| 2.1.4 | Opción para limpiar el cuadro de texto | Josué B | 29/6/20 | 1/7/20 |
| 2.1.5 | Opción para la creación de un archivo HTML | Jose A | 1/7/20 | 4/7/20 |
| 2.1.6 | Opción para traducir las listas | Josué B | 4/7/20 | 6/7/20 |
| 2.2 | Lógica de Negocio |  |  |  |
| 2.2.1 | Función para la ventana de interfaz con TKinter | Jose A | 20/6/20 | 24/6/20 |
| 2.2.2 | Función para obtener texto de un archivo | Josué B | 24/6/20 | 26/6/20 |
| 2.2.3 | Función para separar en listas los tipos de palabras | Jose A | 26/6/20 | 29/6/20 |
| 2.2.4 | Función para reiniciar variables internas | Josué B | 29/6/20 | 1/7/20 |
| 2.2.5 | Función para montar en un HTML con las tablas | Jose A | 1/7/20 | 4/7/20 |
| 2.2.6 | Función para traducir palabras de español a ingles | Josué B | 4/7/20 | 6/7/20 |
| 3 | Actividades de Documentación |  |  |  |
| 3.1 | Documentación Interna del Código |  |  |  |
| 3.1.1 | Determinar E-S-R para cada función | Jose A y Josué B | 19/6/20 | 10/7/20 |
| 3.1.2 | Verificar identificadores y restricciones | Jose A y Josué B | 20/6/20 | 11/6/20 |
| 3.2 | Documentación Externa/Informe Escrito |  |  |  |
| 3.2.1 | Cronograma de Trabajo | Jose A y Josué B | 18/6/20 | 19/6/20 |
| 3.2.2 | Estrategia Metodológica | Jose A y Josué B | 19/6/20 | 20/6/20 |
| 3.2.3 | Tabla de Distribución de Tareas | Jose A y Josué B | 20/6/20 | 21/6/20 |
| 3.2.4 | Bitácora/Diario de Trabajo | Jose A y Josué B | 22/6/20 | 11/7/20 |
| 3.2.5 | Pruebas de Funcionalidad | Jose A y Josué B | 23/6/20 | 11/7/20 |
| 3.2.6 | Manual de Usuario | Jose A y Josué B | 24/6/20 | 11/7/20 |
| 3.2.7 | Análisis de Resultados | Jose A y Josué B | 10/6/20 | 12/7/20 |
| 3.2.8 | Lecciones Aprendidas | Jose A y Josué B | 10/6/20 | 12/7/20 |
| 3.2.9 | Portada, Índice e Introducción | Jose A y Josué B | 11/7/20 | 12/7/20 |
| 3.2.10 | Archivo txt | Jose A y Josué B | 12/7/20 | 12/7/20 |